

INPUT

Publicación práctica
para usuarios de

Sinclair

Febrero 1986 Precio 350 Ptas

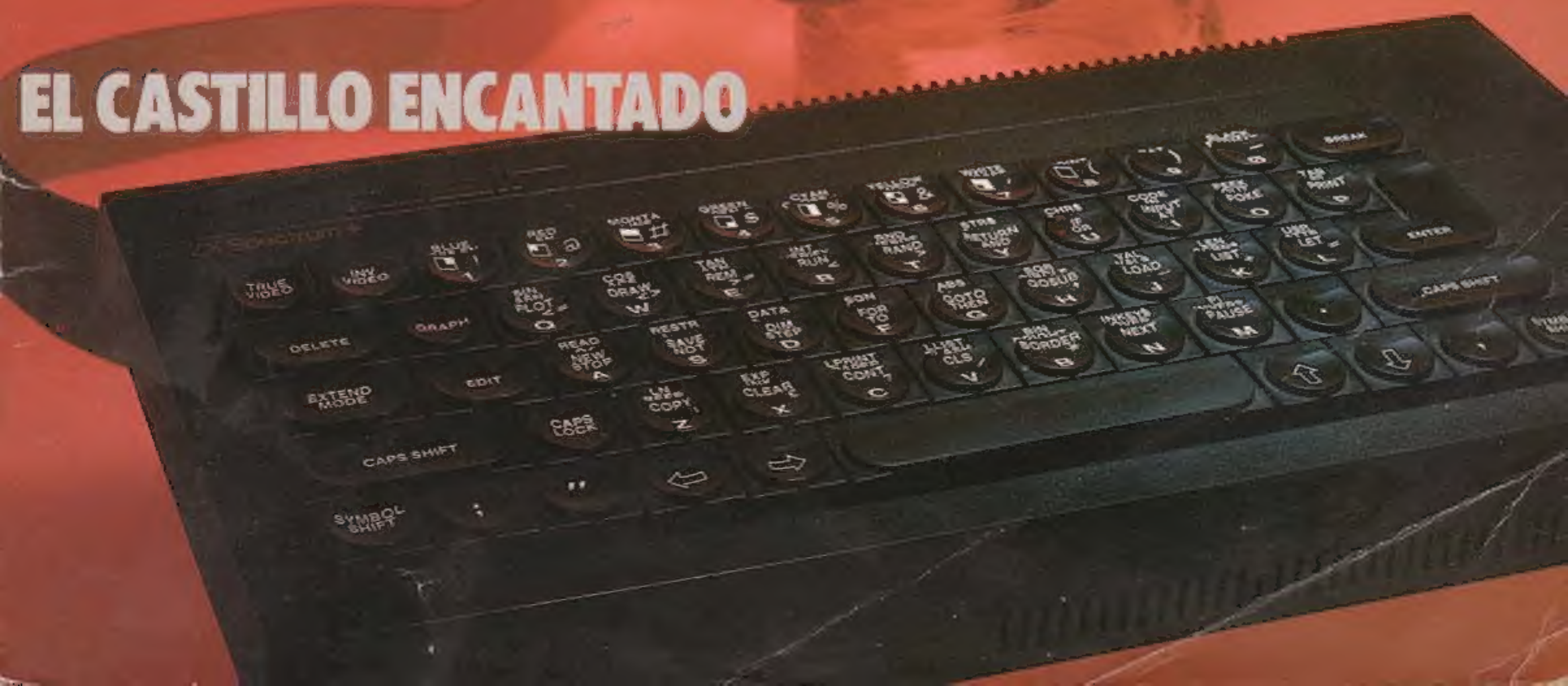
Número 6

Más páginas
de Software

**EXIGELE
A TU BASIC**

**INTELIGENCIA
ARTIFICIAL**

EL CASTILLO ENCANTADO



¡NO LO DECIMOS NOSOTROS!

LOS MEJORES DE INPUT SINCLAIR

PUESTO	TITULO	PORCENTAJE
1.º	<i>The Dambusters</i>	18,8 %
2.º	<i>Profanation</i>	16,1 %
3.º	<i>The way of exploding fist</i> ..	15,3 %
4.º	<i>Super Test</i>	11,8 %
5.º	<i>Basketball International</i> ...	8,3 %
6.º	<i>Hypersport</i>	6,9 %
7.º	<i>Rocky</i>	6,9 %
8.º	<i>Alien 8</i>	6,2 %
9.º	<i>Misión Imposible</i>	5,6 %
10.º	<i>West Bank</i>	4,1 %
		100 %

Para la confección de esta relación únicamente se han tenido en cuenta las votaciones enviadas por nuestros lectores de acuerdo con la sección «Los Mejores de Input».

Enero de 1986

**SI BUSCAS LO MEJOR
LO TIENE**





AÑO 1 NUMERO 6

DIRECTOR:

Alejandro Diges

COORDINADOR EDITORIAL:

Francisco de Molina

DISEÑO GRAFICO:

Tomás López

COLABORADORES:

Antonio Taratíel, Luis R. Palencia,
Francisco Tórtola, Benito Román,
Esther de la Cal, Ernesto del Valle,
Equipo Molisoft.

INPUT Sinclair es una publicación juvenil de
EDICIONES FORUM

GERENTE DIVISION DE REVISTAS:

Angel Sabat

PUBLICIDAD:

Grupo Jota

Madrid: c/ Cuenca, 1, 1.º

Teléf. 253 45 01/02

Barcelona: Avda. de Sarriá, 11-13, 1.º

Teléf. 250 23 99

FOTOMECANICA:

Ochoa, S. A.

COMPOSICION:

EFCA, S. A.

IMPRESION:

Edime, S. A.

Depósito legal: M. 27.884-1985

SUSCRIPCIONES:

EDISA,

López de Hoyos, 141. 28002 Madrid

Teléf. (91) 415 97 12

REDACCION:

Alberto Alcocar, 46, 4.º

28016 Madrid. Teléf. 250 10 00

DISTRIBUIDORA

R.B.A. PROMOTORA DE EDICIONES, S. A.

Travesera de Gracia, 56. Edificio Odiseus,

08006 Barcelona

El precio será el mismo para Canarias que para la

Península y en él irá incluida la sobretasa aérea.

Se ha solicitado el control OJD

INPUT Sinclair es independiente y no está vinculada a

Sinclair Research o sus distribuidores.

INPT no mantiene correspondencia con sus lectores, si bien

la recibe, no responsabilizándose de su pérdida o extravío.

Las respuestas se canalizarán a través de las secciones

adecuadas en estas páginas.

Copyright ilustraciones del fondo gráfico de Marshall

Cavendish, págs. 15, 16, 17, 19, 20, 23, 24, 31, 33, 34,

35, 36, 37, 42, 43, 44, 45, 46.

INPUT sinclair

SUMARIO

EDITORIAL	4
ACTUALIDAD	6
BUZON	7
REVISTA DE HARDWARE	
EL DISKETTE CUMANA	10
EL RATON	26
PROGRAMACION	
TRADUCTOR DE TEXTOS	23
EXIGELE A TU BASIC	15
LAS VARIABLES	42
ENTREVISTA	
MADE IN SPAIN	
CONQUISTA EUROPA	28
PARTICIPA	
EL CASTILLO	40
INTELIGENCIA ARTIFICIAL	
LA I.A. Y LOS	
SISTEMAS EXPERTOS	50
REVISTA DE SOFTWARE	54
EL ZOCO	64
PROGRAMACION DE JUEGOS (COLECCIONABLE)	31
PROYECTA TU AVENTURA (continuación)	
UNA AVENTURA MOVIDA	

NUEVOS TEMAS

A pesar de lo sospechado, el mes de febrero parece continuar en la brecha en cuanto a novedades de *software* se refiere. A la redacción continúan llegando nuevos programas de gran calidad.

Los microordenadores están demostrando ser una útil herramienta para los investigadores y curiosos de las nuevas áreas de la informática. Hasta hace poco, temas tales como las aplicaciones robóticas, la síntesis y el reconocimiento de voz, la inteligencia artificial, los sistemas expertos..., etc. aparecían siempre ligados a los grandes ordenadores. Sin embargo, los microordenadores domésticos pueden ayudarnos a comprender los mecanismos de funcionamiento de las aplicaciones que rozan la ciencia ficción. Resulta obvio que la mayor potencia y velocidad de proceso y las ingentes cantidades de memoria central son las principales bazas de los hermanos adultos del micro, pero en los próximos números de INPUT trataremos de acercar esas palabras grandilocuentes a todos vosotros.

A lo largo de la corta existencia de la revista os hemos invitado a participar con vuestras ideas y cola-

boraciones. Como estímulo complementario venimos ofreciendo tres premios en material informático sorteables entre los participantes del mes. Pues bien, en la próxima edición publicaremos el nombre de los tres ganadores de los tres primeros meses. Así que estar atentos.

Una aclaración, hemos recibido alguna que otra carta de lectores que nos «han pillado» en *off-side* con las páginas centrales de la revista. En realidad la cuestión es otra. Las ocho páginas del cuadernillo central forman un coleccionable, que pretende enseñar el lenguaje BASIC y, en un próximo futuro, código máquina a través de la programación de juegos, ganando en complejidad a medida que avanzamos. El fin último es que dichas páginas sean extraídas y encuadradas con las tapas que hemos preparado. Por ello es lógico que la numeración de las páginas dé un salto al llegar a este punto e, incluso, que el texto de un capítulo necesite más espacio del disponible, continuando en el siguiente número.

Una recomendación: Seguir enviando vuestras colaboraciones inéditas.

LOS MEJORES DE INPUT

Hemos pensado que es interesante disponer de un *ranking* que ponga en claro, mes a mes, cuáles son los programas preferidos de nuestros lectores. Para ello, es obligado preguntaros directamente y tener así el mejor termómetro para conocer vuestras preferencias. Podéis votar por cualquier programa aunque no haya sido comentado todavía en INPUT.

El resultado de las votaciones será publicado en cada número de INPUT.

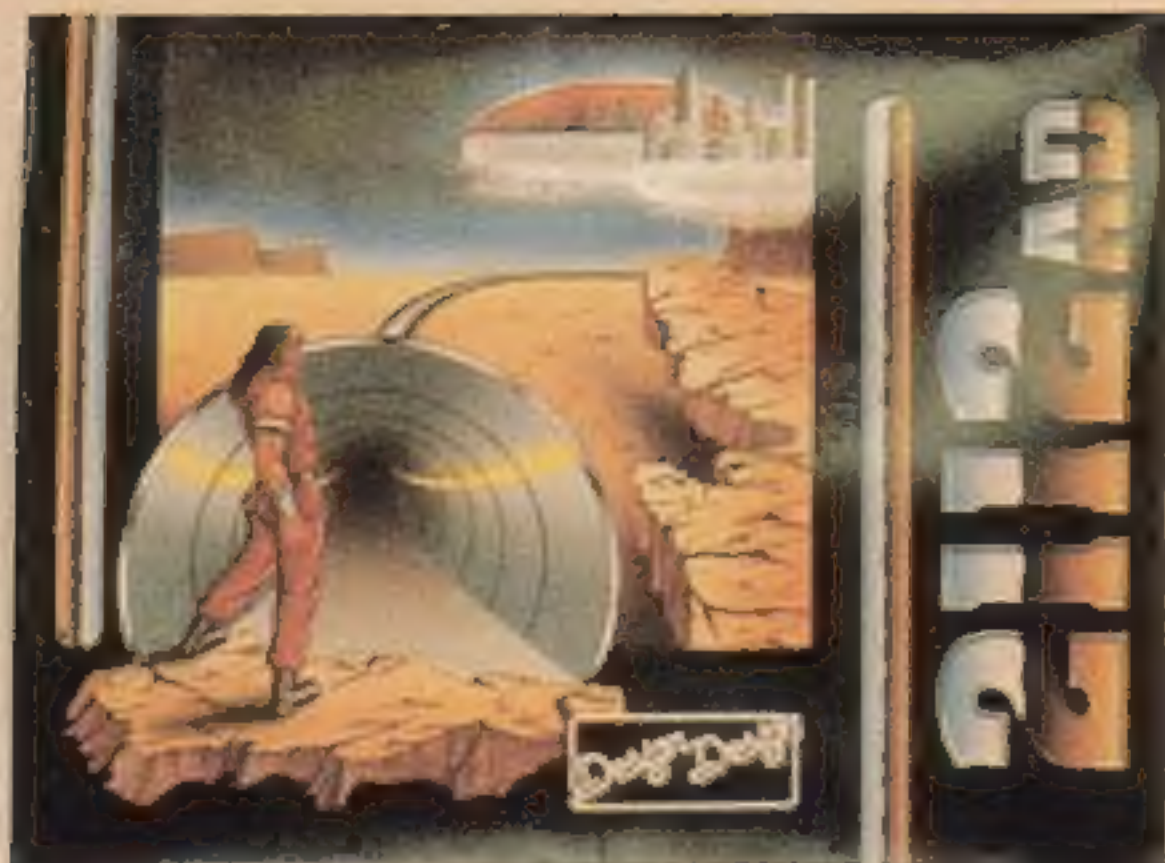
Entre los votantes sortearemos 10 cintas de los títulos que pidáis en vuestros cupones.

Nota: No es preciso que cortéis la revista, una copia hecha a máquina o una simple fotocopia sirven.

Enviad vuestros votos a: **LOS MEJORES DE INPUT** Alberto Alcocer, 46 - 4.º B. 28016 Madrid

ELIGE TUS PROGRAMAS

Primer título elegido	Segundo título elegido
Tercer título elegido	Programa que te gustaría conseguir
Qué ordenador tienes	Nombre
1.º Apellido	2.º Apellido
Fecha de nacimiento	Teléfono
Dirección	Localidad
Provincia	



2112: Nuevo. De Design Desgin.

Con este juego debes usar toda tu inteligencia para vencer las dificultades que te ofrece. Es increíble la velocidad con que los gráficos van cambiando en la pantalla.

Spectrum 48 ó 128 K. P.V.P. 1.800



Forbidden Planet. De Design Design. *Nuevo.*

Este juego fabuloso es tan misterioso que debes descubrirlo por ti mismo. ¡Vive las aventuras!

Para Spectrum 48 y 128 K. P.V.P. 1.800

Wizard Lair: de Bubble Bus.

Te encuentras atrapado en una cueva muy profunda peligrando tu vida. Tienes que buscar la salida venciendo muchos obstáculos.

Spectrum 48 y 128 K.

Commodore 64 y 128 K. Amstrad CPC. P.V.P. 1.900



Starquake: de Bubble Bus. *Nuevo Hit.*

Tu misión es encontrar el camino de Starquake y desactivar los núcleos de los planetas antes de que el mundo estalle. ¡Más de 400 pantallas!

Para Spectrum 48, 128 K.

Próximamente Amstrad CPC. P.V.P. 2.100



Hi Rise: de Bubble Bus. *Nuevo.*

Tú eres el jefe de una empresa de construcción. Tus trabajadores no están contentos contigo. Tienes que escaparte para evitar enfrentamientos.

Para Amstrad CPC. P.V.P.

2.100



Ahora exclusiva para España **British Soft.**

Instrucciones en castellano. Importados de Inglaterra. Todos los precios incluido IVA. Tiendas y distribuidores. Tel. (965) 26 35 93. Pedidos contra-reembolso. Tel. (965) 26 35 93. Disponible directamente por British Soft o en los mejores establecimientos de software.

British Soft. Rocafel, 19. Albufereta (Alicante)

Distribuidor en Madrid:

Computique

Embajadores, 90

28012 Madrid

Tel. (91) 227 09 80



Vectron: de Insight. *Nuevo Hit en Inglaterra.* Vas a ser transportado a un extraño ordenador que genera el universo. Tienes que luchar contra el robot. Rom, tanques y Randomiser. Para Spectrum 48 y 128 K. P.V.P. 1.800

Jep Set Willy II: de Software Projects.

Ahora una segunda versión del famoso héroe Willy. Más de 130 pantallas. Otro hit de la misma casa que Manic Miner. Para Spectrum 48 y 128 K. Amstrad CPC y MSX. P.V.P. 1.800



NUEVO REPRESENTADO

Ultimate es uno de los productores británicos de software más implantados en el mercado del Spectrum. Suyos son títulos de la categoría de sir Athur Pendragon.

Hace pocos días ha pasado a ser otro de los representados por la firma distribuidora Erbe, lo que sin duda contribuirá a que los nuevos programas lleguen con mayor antelación a nuestros comercios.

MAS MEMORIA PARA LOS MICROS

El Coadex es la mayor feria de informática y se celebra anualmente en Las Vegas (EE.UU.). Allí dos firmas japonesas, Hitachi y Toshiba, acaban de presentar sus modelos de almacenamiento masivo en disco (Compact Disk) capaces de guardar hasta 3.600 Megabytes una y 2.600 Megabytes la otra. Sorprendente!!!. En su momento, cualquier microordenador podrá disponer de la información depositada en uno de esos

discos ópticos de 12 pulgadas de diámetro. La lectura se efectúa por medio de un diminuto sistema láser. La principal desventaja de estos sistemas es que actúan como ROM, es decir únicamente puede leerse la información depositada a priori. Varias firmas japonesas presentaron otros sistemas de disco compacto que pueden grabarse una sola vez, siendo una especie de PROM (ROM programable).

LLEGO LA AMNISTIA

En estos tiempos en los que parece que la persecución a los "piratas" del software se recrudece por momentos, fundamentalmente en los países anglosajones, resulta curioso saber que una casa importante se inclina por la amnistía.

Micropro, una importante firma de gran tradición en la esfera de los microordenadores profesionales ha decidido perdonar a los usuarios ilegales de su popular tratamiento de textos Wordstar.

Pero todo tiene un precio. Pagando unas 9.000 pts. el usuario consigue las bendiciones del fabricante y tiene acceso a las posibles actualizaciones y mejoras que el paquete pueda sufrir en el futuro.

El número de copias piratas de este programa se ha evaluado en un millón y medio en todo el mundo. Esta iniciativa podría ser un primer paso para convencer a los usuarios de que los productores de software toman una postura seria hacia todos sus usuarios. De cualquier modo, aunque no por el camino deseado, conseguirán ampliar su cartera de clientes.

OTRA VEZ AL CINE

Donark, firma de software que adquiere popularidad en su día por el programa basado en la película "Panorama para matar", ha lanzado un nuevo título con tema cinematográfico: "Viernes 13".

El juego se centra en proteger a los quinceañeros de las garras del malvado Jason, que con hacha y sierra mecánica los atacará sin piedad. Desconoces dónde se encuentra el villano de la máscara de

hockey, pero cuentas con diez armas y cinco niveles para cazarle. Es un juego de gran suspense.

La cinta viene acompañada por dos cápsulas de sangre artificial y existirá la posibilidad de ganar un monitor con sólo reconocer a qué corresponden los sonidos grabados en la cinta.

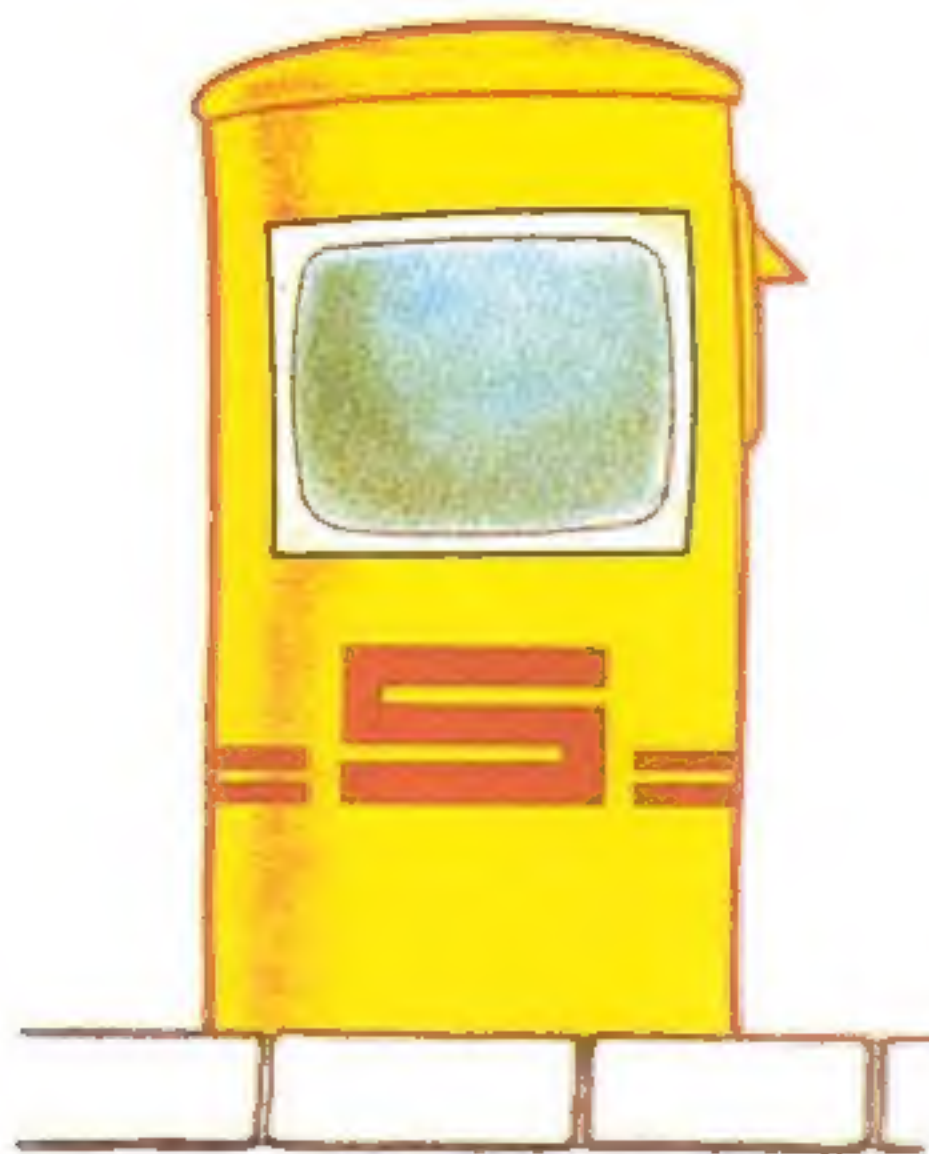
El sonido incluido en el programa contribuye a darle mayor dramatismo. La casa Serna lo importará en nuestro país.

HERBERT EN LA BRECHA

El simpático Herbert vuelve a las andadas, dispuesto siempre a la aventura. Micro Gen lanza un nuevo juego titulado Tres semanas en el paraíso. En las islas Silly se encuentran de repente Herbet y Wilma, la esposa de Wally. Las pantallas confirman ser incluso de mayor calidad que las de episodios anteriores. En ellas aparecen elefantes, leones, murciélagos y otros animales exóticos.

CONCURSO WINTER GAMES

Compulogical puso en circulación recientemente una versión para Spectrum del popular Winter Games, desarrollado en principio para Apple, Atari y C-64. Ahora esta firma ha decidido añadirle emoción al competitivo juego y en breve convocará un concurso de habilidad para usuarios de Spectrum. En el próximo número ampliaremos esta información.



EL BUZON DE INPUT

El Spectrum 128 no es otra cosa que un **Plus** reconvertido por el equipo técnico de **Investrónica**, importador nacional de la marca. Por tanto, sí es factible su conversión, pero para ello es preciso que algún taller disponga de los suministros adecuados y esté dispuesto a realizar las transformaciones o en su defecto que aparezca un kit de piezas. Esta firma es quien mejor te podrá informar, aunque cabe la posibilidad de que algún fabricante británico se decida a hacerlo, igual que en su momento fabricaron otros accesorios para este ordenador.



Quisiera que publicaseis una rutina que me permita alinear por la derecha las cifras que me va produciendo un programa.

Pedro Navas. Sabadell. Barcelona.

Lo más cómodo consiste en traducir el valor a una cadena y entonces manipularla. Esta corta rutina sirve a ese propósito:

El 8 que aparece en la línea 1040 puede ajustarse a la máxima cantidad de cifras que preveas vas a utilizar.



En la revista n.º 4 pedís opiniones sobre el cuadernillo del mismo número, pues la mía es que es algo fabuloso y desearía que se pudiese repetir en más de una ocasión.

Quisiera hacer una pregunta y es la siguiente: ¿En un radio-cassette stereo se pueden leer programas y grabarlos?, o ¿si el mismo aparato poniéndolo en mono esto mismo se podrá hacer?

Ismael Alfarp. Valencia.

Te agradecemos las loas que dedicas al cuadernillo. Sabemos que los lectores estáis un poco saturados de listados de programas y por

eso **INPUT** va en una línea más didáctica. Para ofrecer listados existen otras publicaciones que cumplen muy bien este cometido. De todas maneras, seguiremos ofreciéndolos obsequios como el cuadernillo y otros más interesantes por venir.

Un cassette estereo se diferencia, en lo que te interesa, de uno normal por el hecho de que el lugar destinado a la pista de datos en un modelo monofónico se divide en dos, una para cada canal. Por ello existe la posibilidad de que una cinta de software comercial leída en un estereo pueda producir problemas de carga.

Por otro lado, si has grabado un programa propio utilizando uno de los canales, no debería en principio surgir ninguna pega cuando lo vuelvas a cargar.



Quisiera preguntaros si tenéis pensado hacer unas «tapas» para encuadernar las revistas, ya que como son tan «bonitas» da pena tenerlas por ahí. ¿Me entendéis?

Yo había pensado en ese sistema de varillas y no hace falta encuadernarlas y quedan muy bien y ya de paso, si las vais a sacar, que sean a precio lo más bajo posible.

Fernando Castán. Zaragoza.

Efectivamente, tenemos pensado sacar tapas para las revistas en el momento en que se aconseje el número de ejemplares.

El precio queremos que sea lo más bajo que los costos de producción y envío permitan. De todas formas, muchas gracias por tus sugerencias.



Desaría que me respondieseis a las siguientes preguntas: ¿Cualquier teclado semiprofesional tiene entradas para joystick? De no ser así, ¿el teclado de Indescomp tiene?

Tengo un ZX Spectrum y me gustaría conocer la siguiente información. Quisiera saber cuál es la memoria ROM del Plus y cuántos K tiene. También quisiera conocer vuestra opinión con respecto al teclado del Plus y el Indescomp (cuál es mejor o cuál tiene más comodidades) y si merecería la pena cambiar un teclado por el otro.

Por último, ¿se podría transformar el ZX Spectrum en el 128 K?

Alejandro Verde. Guadalajara.

La memoria ROM del **ZX Spectrum Plus** es la misma que la utilizada por el modelo anterior (es fundamentalmente el mismo) y es de 16 Kbytes. El teclado del **Plus** es igual que el utilizado por el **Spectrum** antiguo, la diferencia principal estriba en que se han sustituido las teclas de goma por otras de plástico rígido. Esto tenía sentido cuando los teclados eran caros para los fabricantes, pero en la actualidad no es ese el caso.

Para más información sobre el otro teclado que mencionas, en el número 4 de la revista proponíamos una aplicación profesional en la que éste era parte del proyecto. Sus teclas llevan debajo pulsadores y por tanto se acerca más al tipo de teclas que emplean otros ordenadores. Sin embargo, no debemos aconsejarte, tú deberás tomar la determinación cuando puedas establecer cuál te agrada más en la tienda de microinformática.

EL BUZON DE INPUT

Si conecto cualquier interface al Spectrum, ¿podré luego desconectarlo sin tener ningún problema posterior?

En algunas fuentes de información para el Spectrum me he enterado de que no se podía hacer nada para que un programa en C.M. se autoejecutase sin un anterior programa en BASIC. Sin embargo, he notado que en Maziacs y Pinball, entre otros, se autoejecutan solos. ¿Cómo es esto posible?

¿Cuál es el nemotécnico para producir sonido en C.M.?

Jesús López. Alcalá de Henares.

Un teclado como accesorio independiente del ordenador no tiene porque incorporar entradas para joystick. Es más, para el Spectrum existen distintas posibilidades de joystick (programable, Kempston, etc.). Sin embargo, pueden incorporar teclas separadas dispuestas en cruz, que son duplicación de las numéricas que suelen emplearse

con el mismo fin, pero que resultan incómodas durante los juegos. El teclado de Indescomp incorpora dicha réplica de teclas, aunque esta firma nacional también fabrica una excelente gama de interfaces para joysticks.

El Spectrum puede aceptar cualquier interface específicamente diseñado para él. No obstante conviene tomar una simple precaución para evitar desagradables sorpresas: desconectar la alimentación del ordenador cuando se enchufe o desenchufe el interface al slot de expansión. En caso contrario es muy probable que se produzcan averías, sobre todo al desenchufarlo mientras el ordenador está alimentado. Por lo demás no hay contraindicaciones.

Es procedimiento normal utilizar una breve rutina en BASIC para cargar y autoejecutar un programa escrito en código máquina. El BASIC no es más que un amplio conjunto de instrucciones escritas en código máquina, por lo que es perfecta-

mente factible desarrollar un programa en c.m. que utilice a otras rutinas en c.m. propias del BASIC para obtener mayor grado de protección contra copias. Su manejo requiere sólidos conocimientos del c.m. y las interioridades del ordenador.

No existe una instrucción simple del juego del microprocesador Z80 que te permita controlar el altavoz del Spectrum. Este diminuto productor de sonido está conectado a la patilla 28 del chip ULA (igual que los terminales de micrófono y auricular para el cassette) y éste le controla directamente enviando señales eléctricas. Existen posibilidades de control desde código máquina, pero utilizar el comando BEEP del BASIC resulta más lógico por la mayor sencillez y menos dependencia crítica del tiempo. De todas formas existe una interesante y prolija fórmula de producir sonidos y valerte de las interrupciones para que el ordenador pueda hacer más cosas casi simultáneamente.

17%

de descuento

Suscríbase ahora a INPUT!!

Por sólo **290 Ptas.** ejemplar, y recibidos todos cómodamente en su hogar...

PRECIO DE CUBIERTA PTAS. ~~350~~
MENOS:
17% de descuento al suscriptor PTAS. (60)

USTED PAGA SOLO PTAS. **290**
POR EJEMPLAR

SUSCRIPCION ANUAL = 11 EJEMPLARES

~~3.850 Ptas.~~

(660 Ptas.)

3.190 Ptas.

Entrega a domicilio GRATIS

INPUT le proporciona
INFORMACION... DIVERSION...
...FORMACION (un curso completo de programación)...
...LA POSIBILIDAD DE MEJORAR su NIVEL PROFESIONAL...
EL NIVEL DE LOS ESTUDIOS...

...Descubra el mundo de la informática...

...Aprenda a programar con facilidad...

...Diviértase con los ordenadores...

...Esté siempre al día...

Recorte y envíe este cupón de inmediato a EDISA, López de Hoyos, 141-28002 Madrid, o bien llámenos al Telf. (91) 415 97 12

BOLETIN DE SUSCRIPCION

SI, envíeme INPUT SINCLAIR durante 1 año (10 ejemplares + el extraordinario de verano), al precio especial de oferta de **3.190 Ptas. AHORRANDOME 660 Ptas.** sobre el precio normal de portada de 11 ejemplares sueltos. (Por favor cumplimente este boletín con sus datos personales e indiquenos con una (X) la forma de pago por usted elegida, métele en un sobre y deposítelo en el buzón más próximo).

NOMBRE _____ APELLIDOS _____
DOMICILIO _____ C/P _____ PISO _____ ESCALERA _____ COD. POSTAL _____
POBLACION _____ PROVINCIA _____ TEL. _____
PROFESION _____

FORMA DE PAGO ELEGIDA: Reembolso ☐ Domiciliación Bancaria ☐
Talón nominativo que adjunto a favor de EDISA ☐

INSTRUCCIONES DE DOMICILIACION BANCARIA (si es elegida por usted)

Muy señores míos, _____ de _____ de 19____
Les ruego que, con cargo a mi cuenta nº _____ atiendan, hasta nuevo aviso, el pago de los recibos que les presentará Editorial PLANETA-AGOSTINI a nombre de _____
BANCO/C de AHORROS _____
DIRECCION _____
FOMA _____

Ahora exclusiva para España **British Soft.**

Instrucciones en castellano. Importados de Inglaterra. Todos los precios incluido IVA. Tiendas y distribuidores. Tel. (965) 26 35 93. Pedidos contra-reembolso. Tel. (965) 26 35 93. Disponible directamente por British Soft o en los mejores establecimientos de software.

British Soft. Rocafel, 19. Albufereta (Alicante)

Distribuidor en Madrid:

Computique

Embajadores, 90

28012 Madrid

Tel. (91) 227 09 80



Leuc: de THOR
Computer Software.

Nuevo. El juego más reciente de Odin.

Terminado de fabricar en enero. Para Spectrum 48, 128 K. Commodore 64 ó 128 K. P.V.P. 2.300



Robin The Wood: de
Odin Computer Graphics.
Crash Mash nuevo para
Amstrad.

¿Puedes tú,
igual que Robin ganar el
trofeo más preciado,
como es la flecha de
plata? Para Spectrum 48,
128 K. Commodore 64 ó
128 K. Amstrad CPC.
P.V.P. 2.300



The Arc Of Yesod:
Nuevo Crash Mash. De
Thor Computer Software.
este juego

es la continuación de
Nades Of Yesod.
Descubrirás un mundo
galáctico verdaderamente
sorprendente. Spectrum
48 ó 128 K. Commodore
64 ó 128 K.
Próximamente Amstrad
CPC. P.V.P. 2.300



Marsport: de Gargoyle
Game. Crash Mash.

Este juego es la
primera parte de la
trilogía. «El asedio de la tierra».
Las críticas de
Microhobby «Marsport es
un excelente juego sin
igual». Amstrad CPC,
Spectrum 48 ó 128 K.
P.V.P. 2.300



Four Pack: Amstrad
CPC. De Software
Proyectos cuatro
fantásticos juegos: Manic
Miner, Binky, Jep Set
Willy. Kars. Treasure
Hunt. En disquette y
ahora también en
cassette, constituyendo
una novedad
sorprendente.
Disquette P.V.P. 4.000
Cassette P.V.P. 2.300



Sweevos World: de
Gargoyle Game. Nuevo
Hit. Es la más divertida
aventura de dibujos
animados en la que tú
puedes participar. Para
Amstrad CPC, Spectrum
48 ó 128 K. P.V.P. 1.900

DISKETTE CUMANA E INTERFACE BETA-DISK

Hay dos razones clarísimas, aunque no las únicas, para utilizar un *diskette*: Velocidad de lectura/escritura y acceso aleatorio.

Estos dos requerimientos no pueden satisfacerse con un *cassette*, o al menos con una de tipo doméstico.

La velocidad resulta imprescindible en la práctica cuando han de manejarse datos o programas que por su amplitud no pueden estar residentes en memoria y ha de hacerse frecuente uso de algunas de las partes, almacenadas lógicamente en un medio exterior de mayor capacidad.

Sin lugar a duda, la posibilidad de acceso aleatorio es decisiva. Con el *cassette* sólo pueden realizarse accesos secuenciales, mientras con el *diskette* podemos grabar (SAVE) o leer (LOAD) programas o datos en cualquier orden.

La seguridad es un factor importante. Los *diskettes* son mucho más fiables que los *cassettes* (durante el formateo inicial se descartan las zonas defectuosas) y de comportamiento más homogéneo. Todos sabemos que no siempre podemos hacer uso de un programa que nos han dejado y que «rueda perfectamente en el equipo de quien nos lo ha dejado». Unas veces la culpa la tiene la velocidad de giro, otras la altura de las cabezas y quizá otras los niveles de grabación. Con un *diskette* es mucho más improbable que ocurra algo de esto.

Hasta ahora, la verdad es que en España no es fácil encontrar una amplia gama de *diskettes* adaptables al ZX Spectrum. A lo sumo 3 ó 4 marcas. Nosotros llevamos unos cuantos meses utilizando el conjunto Beta-Disk (*interface*)/Cumana (unidad de discos con excelentes resultados y a él nos vamos a referir en este artículo).

Desconocemos la situación comercial de este conjunto en el momento

actual, pero tenemos noticias que hay vendidos aquí unos trescientos, cantidad que parece interesante. Para aquellos que deseen aventurarse en este campo les indicaremos que su precio es de unas 28.000 ptas. el *interface* y 60.000 ptas. la unidad de 400 k. Su distribuidor era la firma Silog (Gerona).

UNIDAD DE DISKETTE CUMANA

El equipo viene provisto de un documentado manual de instrucciones (en inglés), aunque realmente no es de mucha utilidad en nuestro caso. La razón es que está enfocado a su conexión al microordenador BBC. No obstante, nos detendremos en aquellos aspectos eléctricos y mecánicos que pueden resultarnos de interés. La parte de *software* la veremos con todo detalle en el manual del *interface Beta-Disk*.

La verdad es que podríamos haber elegido cualquiera de las muchas unidades que existen en el mercado americano, inglés o japonés y que son compatibles con una gran mayoría de los microordenadores, como por ejemplo Shugart, Opus, etc.

Centrándonos en el Cumana diremos que nuestra versión está preparada para ser conectada a la red alterna de 200 v. Algunos micros (por ejemplo el BBC) disponen de fuente de alimentación incorporada y, por lo tanto, se requiere otro modelo algo más sencillo (y barato). No olvidéis este detalle al comprarlo.

En cuanto a la capacidad de almacenamiento básicamente hay dos modelos: 40 pistas y 80 pistas (48 tpi y 96 tpi respectivamente. tpi = *track per inch* = pistas por pulgada) en formato de 5-1/4" (también pueden elegirse de 3-1/2", con las mismas capacidades y mucho más compactos, a cambio de unos *diskettes* menos populares y de

mayor precio). En todos los casos el formato es de 10 sectores por pista, por lo que sólo se requieren *diskettes* de «densidad sencilla».

En nuestro caso utilizamos un modelo mixto bastante versátil ya que podemos trabajar a simple o doble cara eligiendo funcionar, con sólo oprimir una palanca, a 40 u 80 pistas. Podemos pues leer o editar discos tanto de una o dos caras, como 40 u 80 pistas.



O lo que es lo mismo de 100 k a 400 k.

No obstante, debe tenerse la precaución de utilizar un disco totalmente «limpio» cuando se haga la grabación a 40 pistas con una unidad 40/80, si ha de ser leído por otra unidad preparada para sólo 40 pistas.

En un disco de sólo 40 pistas por cara, la cabeza lectora comienza su exploración por la pista 0 (la más exterior), avanzando hasta la 39 (más interior), a razón de 48 tpi (pistas por pulgada). La capacidad total de almacenamiento es por tanto de 100 Kbytes (40 pistas \times 10 sectores \times 256 bytes por sector). En la pista 0 se almacena el fichero que contiene información sobre el contenido del disco.

En los discos de 80 pistas la situa-

ción es análoga pero con doble número de pistas (una cara) o cuádruple número (doble cara). La anchura de cada pista es por tanto «la mitad», con la salvedad que se encuentran desplazadas de forma que las pistas pares queden centradas respecto a las pistas en formato 40.

Un disco grabado en una unidad de 40 pistas puede ser manejado perfectamente por uno 40/80 en modo 40 ya que aunque la banda magnética utilizada es la mitad de ancha no presenta problema alguno.

Por el contrario, hay que tener cuidado en utilizar un disco «limpio» cuando se haga una grabación en la de 40/80 en modo 40 para ser leída por la unidad de sólo 40 pistas. Si en las pis-

tas impares quedan restos de otras grabaciones el resultado será algo ininteligible, ya que al ser las pistas de doble anchura la información leída será una mezcla de la pista correspondiente y las dos semipistas adyacentes.

Por lo que respecta al manejo de la unidad no hay problema especial. Una vez conectado tal como indica la Fig. 1, sólo resta introducir el *diskette* por la ranura frontal («boca») y accionar la palanca de cierre (ver Fig. 2). Para retirar el *diskette* la operación es justo la opuesta.

En la parte frontal existe una luz roja (diodo LED) que se enciende cuando la cabeza está leyendo o escribiendo datos.

INTERFACE BETA-DISK

Está fabricado por la casa **Technology Research Ltd** de Inglaterra.

El *interface* viene preparado para controlar hasta 4 unidades, (A, B, C y D) es decir, hasta 1,6 Mega bytes, lo cual empieza a ser ya una cifra respetable.

El *software* del sistema operativo D.O.S (*Disk Operating System*) está preparado para situarse a partir de la posición de memoria 15360. Esta zona está situada en la ROM pero no se utiliza el **ZX Spectrum** por lo que no se crea ningún conflicto y además emplea muy pocos bytes de la RAM.

El paso al D.O.S desde BASIC se consigue mediante **RANDOMIZE USR 15360** **[ENTER]** (el puntero cambia a «A>»). Para volver al BASIC desde el D.O.S basta con pulsar **RETURN** **[ENTER]** (el puntero cambia al cuadrado parpadeante).

Si se desea acceder al D.O.S desde una instrucción de un programa BASIC, de forma que una vez ejecutado el comando correspondiente de control de disco se continúe automáticamente con el programa BASIC, debe teclearse: «**RANDOMIZE USR 15363: REM: 'comando correspondiente'**». No puede utilizarse más de un comando a la vez, y el conjunto debe figurar como una instrucción aislada.

Por ejemplo, para pasar un progra-



ma a disco y *cassette*, el programa sería:

Con el *interface* se suministra también un *diskette* con una serie de programas de utilidades básicas:

- BACKUP** : Duplica todo el contenido de un disco para obtener copias de seguridad.
- COPY** : Copia de ficheros desde una unidad a otra. El puntero pasa a A*.
Ej.: A* COPY «C: prog1», «A: prog3».
- SCOPY** : Copia de ficheros de un *diskette* a otro pero disponiendo sólo de una unidad. El puntero pasa a A1 COPY «tabla 2» CODE.
- FORMAT** : Formateo de *diskettes* a 40 u 80 pistas con una o dos caras.

Durante el proceso del formateo se tiene la posibilidad de dar un nombre al disco y dotarle de una clave. El nombre aparece siempre que se solicita el catálogo y puede resultar de gran ayuda si, por ejemplo, los tenemos clasificados por temas, clientes, etc. El uso de una clave es imprescindible cuando se quiere evitar que manejen el contenido del disco personas no autorizadas.

La clave de acceso puede cambiarse mediante el comando **USR**.

Cuando se dispone de más de una unidad debe indicarse lógicamente a cuál nos dirigimos en cada caso. Con **LOAD** «B: prog» cargaremos el programa «prog» desde la B y con «C: prog» lo haremos de la C. Si no se especifica ninguna unidad el ordenador tomará el establecido «por defecto».

De no indicar nada, la elegida por defecto es la A, pero puede cambiarse por la B, C, o D mediante *«B:», *«C:» o *«D:» respectivamente.

Por último haremos un breve comentario de cada uno de los comandos del D.O.S

- CAT** : Presenta en pantalla el nombre del *diskette*, el número de ficheros en uso y cancelados, cada uno de ellos (con nombre, tipo y longitud) así como la capacidad aún disponible.
- Ej.:
Title: juegos
4 File(s)
2 Del. File(s)
A: carta <D> 5
A: lista <D> 2
A: instr. <C> 4
A: juego 1 3
370 Free

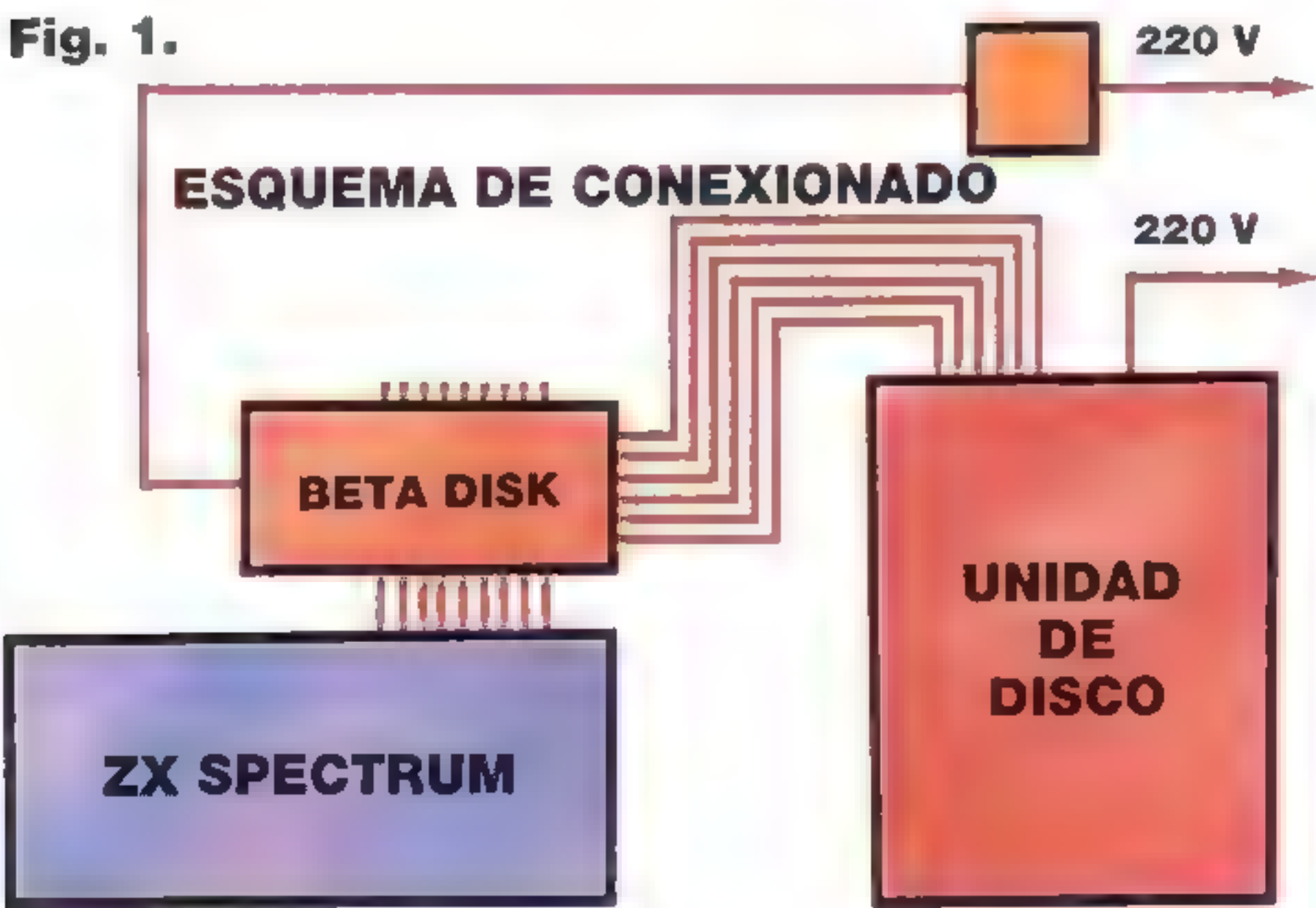
Hay 4 ficheros activos y 2 cancelados (pero su espacio aún está reservado).

De los ficheros activos 2 son de código máquina (5 y 4 sectores respectivamente), 1 para fichero de una matriz de datos (2 sectores) y otro para un programa BASIC (3 sectores).

Hay ocupados además 10 sectores para el fichero del catálogo general del disco (pista 0), por lo que sólo quedan 370 sectores libres.

- ERASE** : Elimina o borra un fichero del disco, aun cuando el espacio correspondiente no puede utilizarse de momento Ej.: **ERASE** «B: prog».
- LOAD** : Equivalente al **LOAD** de *cassette*.
- MERGE** : Equivalente al **MERGE** de *cassette*.
- MOVE** : Permite compactar los ficheros, eliminar los espacios muertos producidos durante las cancelaciones (**ERASE**). De no utilizar **MOVE** pudiera ocurrir que existiese suficiente espacio vacío para alojar un nuevo fichero pero no pudiéramos conseguirlo por no haber un «hueco» único de dimensiones suficientes
Se requieren 4 K de memoria RAM libre para este comando.
- NEW** : Sirve para cambiar el nombre a un fichero.
Ej.: **NEW** «C: nombre nuevo», «nombre viejo» CODE, cambia el nombre a un fichero de C/M en la unidad C.
- PEEK** : Sabiendo como está archivado un fichero determinado (a través de **CAT**) podemos pasar a memoria uno cualquiera de los sectores (256

Fig. 1.



EL IVA
LO PAGA MICRO-1

MICRO-1

Duque de Sesto, 50. 28009 Madrid
Teléfs.: (91) 275 96 16/274 53 80
MICROLID: Gregorio Fernández

SOFTWARE:

¡¡ENHORABUENA, compra 2 programas por el precio de 1!!

SUMMER GAMES	2.190 ptas.	SUMMER GAMES II	2.190 ptas.
IMPOSIBLE MISSION	2.190 ptas.	911 TS	1.750 ptas.
ASTROCLONE	1.900 ptas.	RAMBO	2.100 ptas.
GYROSCOPE	1.900 ptas.	MAPGAME	2.700 ptas.
SABOTEUR	1.900 ptas.	YIER KUNG FU	1.900 ptas.
WEST BANK	1.950 ptas.	CAMELOT WARRIOR	2.100 ptas.
MILLION (4 JUEGOS)	2.500 ptas.	CRITICAL MASS	1.950 ptas.
DAMBUSTER	2.100 ptas.	SUPER TEST	2.100 ptas.
SGRIZAM	1.950 ptas.	POPEYE	1.875 ptas.
BEACH HEAD II	2.100 ptas.	TOMAHAWK	2.495 ptas.
OLE TORO	2.100 ptas.	NIGHTSADE	1.950 ptas.
LOTERIA PRIMITIVA	1.750 ptas.	SUPERTEST	1.695 ptas.

POR CADA CINTA QUE NOS PIDAS, RECIBIRAS GRATIS UNO DE LOS SIGUIENTES PROGRAMAS:

FRANKIE G. TO HOLLYWOOD
SOUTHERN BELLE
DRAGONTORC
MAPSNACH

DUMMY RUN
EXPLODING FIST
BABALIBA
VIDEOLIMPIC

BOUNTY BOB
TAPPER
SAIMAZOO
GREMLINS

CONVIERTE TU SPECTRUM A PLUS
¡¡7.990 PTAS.!!

OPUS DISCOVERY
DISKETTE 3.5"
¡¡48.900 PTAS.!!

SERVICIO TECNICO DE REPARACIONES
SPECTRUM. TARIFA FIJA: 3.800 PTAS.

AMPLIFICADOR DE SONIDO
SPECTRUM 2.450 PTAS.

IMPRESORA MARGARITA
¡¡49.900 PTAS.!!

CASSETTE ESPECIAL
ORDENADOR 5.295 PTAS.

LAPIZ OPTICO
¡¡3.680 PTAS.!!

TOSHIBA MSX 64 K
¡¡34.900 PTAS.!!

PRECIOS SUPER-EXCEPCIONALES PARA
AMSTRAD CPC-472 Y CPC-6128
¡¡LLAMAMOS, TE ASOMBRARAS!!

AMPLIACIONES DE MEMORIA
¡¡4.500 PTAS.!!

SPECTRUM PLUS: 31.500
SINCLAIR QL: 68.900

IMPRESORA SEIKOSHA GP-50S
¡¡19.500 PTAS.!!

TECLADOS PROFESIONALES:
—SAGA-1 10.900
—INDESCOMP 13.195

OFERTA IMPRESORAS: TODAS LAS MARCAS
CON UN ¡¡20% DTO. SOBRE P.V.P.!!

OFERTAS JOYSTICK QUICK SHOT
QUICK SHOT I + INTERFACE 3.350
QUICK SHOT II + INTERFACE 3.895
QUICK SHOT V + INTERFACE 4.350

PC-COMPATIBLE IBM 256K
MONITOR FOSFORO VERDE
2 BOCAS DISKETTE 360K
SOLO ¡¡243.900!!

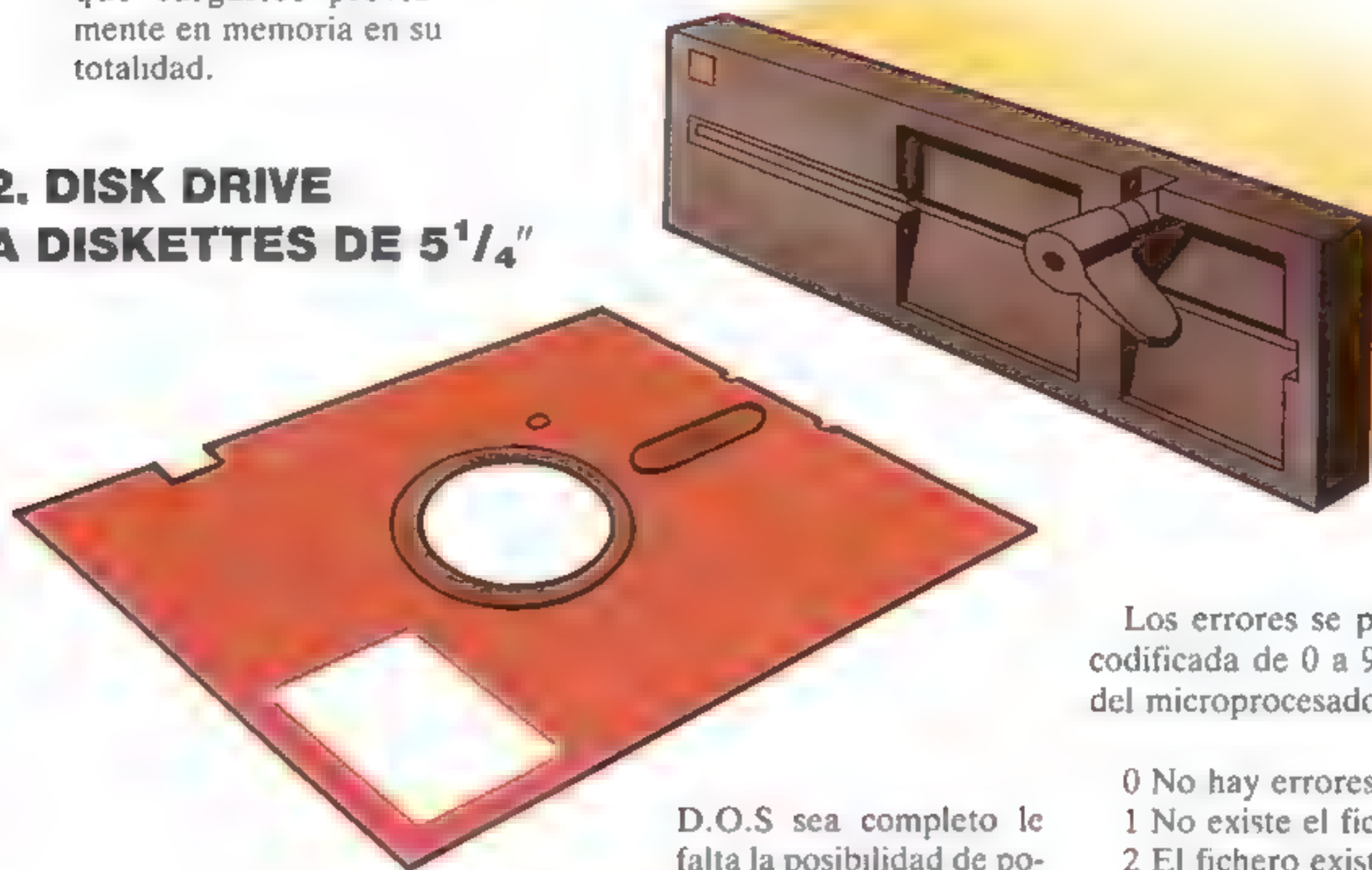
CARTUCHOS MICRODRIVE 495
DISKETTES 5 1/4" 350
QUICK DISK (2.8") 32.875

CINTA C-15 ESPECIAL ORDENADOR 85
INTERFACE CENTRONICS/RS-232 8.495
INTERFACE DOBLE 3.495

Pedidos contra reembolso sin ningún gasto de envío. Teléfs.: (91) 275 96 16 / 274 53 80, o escribiendo a Micro-1
C/Duque de Sesto, 50 - 28009 Madrid

bytes). De esta manera se puede trabajar con datos de ficheros sin tener que cargarlos previamente en memoria en su totalidad.

Fig. 2. DISK DRIVE PARA DISKETTES DE 5 1/4"



Los errores se presentan en forma codificada de 0 a 9 en el registro BC del microprocesador Z80A:

- 0 No hay errores.
- 1 No existe el fichero.
- 2 El fichero existe.
- 3 No hay espacio.
- 4 No asignado.
- 5 Sobrepassado el número de registros.
- 6 No asignado.
- 7 Errores de disco.
- 8 Errores de sintaxis.
- 9 No asignado.

Para hacer un uso práctico de esta detección se pasa el contenido del registro BC a una variable cualquiera, por ejemplo, ERROR mediante: LET ERROR=USR 15363:REM: <comando>.

Supongamos que se trata de archivar un programa en disco, sustituyendo a otro del mismo nombre, si lo hay, y con posible escasez de espacio disponible:

Uno de los puntos flacos de este conjunto radica en los posibles falsos contactos en los conectores al moverlo accidentalmente o durante su uso normal. El problema se agrava un poco más al añadir el *interface* de la impresora. Por experiencia, sugerimos fijar el ZX Spectrum y todos los *interfaces* a una superficie rígida.

Por lo demás los resultados obtenidos han sido excelentes.

Ej.: PEEK «B: tren» 30000,10 pasa a la RAM, a partir de la posición 30000, el sector número 10 del fichero «tren» de la unidad B.

POKE : Instrucción complementaria de PEEK, que permite modificar directamente en disco un sector determinado de un fichero.

Ej.: POKE «B: tren» 30000,10 modifica el sector número 10 del fichero «tren» de la unidad B con 256 bytes de la RAM, a partir de la posición 30000.

PEEK y POKE son dos instrucciones muy potentes que permiten usar o modificar datos de ficheros que por su longitud no pueden cargarse en RAM en su totalidad o que no procede hacerlo a fin de ganar velocidad. Con todo, para que este

RUN

D.O.S sea completo le falta la posibilidad de poder acceder a un registro dado de un fichero sin especificar su ubicación en el disco. Technology Research Ltd. ha anunciado esta facilidad para una versión posterior.

SAVE

: Permite cargar un programa de disco y ejecutarlo automáticamente tanto si es BASIC como Código Máquina.
: Comando análogo al de *cassette*. No puede ejecutarse si existe ya un fichero del mismo tipo y con el mismo nombre. De esta manera tenemos la seguridad de no «machacar» inadvertidamente algún fichero válido. Podemos hacer uso de NEW o ERASE si procede.

Cuando en algunas de las operaciones D.O.S se produce un error no aparece ningún mensaje de aviso en la pantalla pero sí internamente. El Beta-Disk es capaz de detectar hasta 8 tipos de errores.

EXIGELE A TU BASIC

HAZ QUE TUS PROGRAMAS
EN BASIC CORRAN MAS
TIEMPO DE EJECUCION
DE LAS SENTENCIAS
ATENCION A LA ESTRUCTURA

Ponte el casco, la visera y las ropas de viaje. Limpia tus bujías y asegúrate de que tu máquina corre con suavidad. Está a punto de bajarse la bandera y ya es tiempo de acelerar tu BASIC.

Al principio, todos los ordenadores parecían rápidos como un rayo, realizando «en un abrir y cerrar de ojos» tareas que a cualquier persona le requerirían mucho más tiempo. Pero prueba a escribir un juego de acción, o hacer que la máquina ejecute una larga serie de cálculos, o un trabajo complejo de clasificación, con el BA-

SIC y te encontrarás con una situación muy distinta. Verás que realmente la máquina tarda en terminar la tarea que se le ha asignado. Puede que muy pronto empieces a quejarte de lo lenta que es.

Los programas que se ejecutan con más rapidez son los que están escritos en código máquina o lenguaje ensamblador, si bien la mayor parte de la gente encuentra más fácil programar en BASIC. Desgraciadamente, un programa en BASIC nunca puede esperar ni siquiera acercarse a la velocidad de un programa escrito en código

máquina, debido a que el ordenador tiene que gastar tiempo traduciendo las instrucciones del BASIC al código máquina. Para ello lleva incorporado un programa especial que se ocupa de este trabajo: el intérprete.

Si tú no quieres escribir programas en código máquina, pero sí apurar al máximo la velocidad de tu máquina, puedes ensayar varios trucos. En primer lugar, procura estructurar adecuadamente tus programas. En segundo lugar, tienes que intentar conseguir que cada línea individual de programa opere a la velocidad óptima. Elige



pues los componentes del BASIC que el intérprete traduce con más rapidez.

Cada máquina tiene sus propias peculiaridades, y en cierta medida, cada programa tiene sus propias exigencias. En consecuencia no existen reglas generales e infalibles para conseguir el programa perfecto. Sólo se pueden dar ciertas indicaciones, y a tí te corresponde escoger los trucos a incorporar a tus programas, ya que el uso de los mismos puede requerir sacrificar otras cosas.

MEDIDA DE TIEMPOS EN PROGRAMAS DE BASIC

Todas las máquinas llevan incorporado un reloj que puede utilizarse para comparar la rapidez de unos programas respecto a otro. Teclea la rutina siguiente, especialmente diseñada para tu máquina, para que puedas ir

viendo por ti mismo en los ejemplos posteriores de este artículo las diferencias entre formas alternativas de programar en BASIC:

```
1 POKE 23672,0:POKE 23673,0:
  POKE 23674,0
100 REM TEMPORIZADOR PARA EL
    SPECTRUM
120 FOR i=1 TO 100
130 GO SUB 200:NEXT i
140 LET b=PEEK 23672+256*
    PEEK 23673+65536*PEEK
    23674
150 PRINT AT 5,5;(b-41)/5;
    " MILISEGS"
160 STOP
200 REM
500 RETURN
```

Observa la sentencia REM en la línea 200, que será utilizada posteriormente para encajar con la parte que se quiera probar. La sentencia REM de la línea 100 forma parte de la corrección de cronometraje y no debe ser omitida.

ESTRUCTURA

Aunque ya hemos hablado bastante en INPUT sobre los programas estructurados, merece la pena insistir en aquellos factores que más influyen en la velocidad de ejecución.

Todas las subrutinas que se utilicen con frecuencia, tienes que situarlas cerca de principio del programa. La razón de esto es que el intérprete empieza a buscar el número de la línea que aparece junto al GOSUB desde el principio. Obviamente, si el número de línea es bajo, será encontrada con más rapidez que si es alto. El ahorro de unos milisegundos aquí y allá puede convertirse enseguida en una contribución significativa en cuanto a velocidad.

Los programas mal pensados que utilizan un laberinto indiscriminado de GOTOs —a los que a veces se llama programas de *spaghetti*— no sólo constituyen una seria amenaza para los que intenten leer el programa, sino que pueden ser un gran obstáculo para la ejecución rápida. Así pues, un programa bien planeado es probable que sea más rápido de ejecución que uno



que has ido modificando una y otra vez sin un plan previo.

MEMORIA Y VELOCIDAD

En general, los programas cortos es muy probable que se ejecuten también más rápidamente. Sin embargo, las tres condiciones principales que debe cumplir un programa —velocidad, claridad y economía de memoria— suelen estar normalmente en conflicto. Por ejemplo, las líneas de programa con muchas sentencias, sirven para ahorrar memoria y aumentar la velocidad, pero pueden hacer que el listado sea más difícil de seguir y depurar.

Un programa diseñado para tener máxima velocidad de ejecución, es posible que aumente mucho en longitud y en uso de memoria. Recíprocamente, algunos programas que son muy efectivos para ahorrar memoria, lo consiguen a expensas de un aumento de velocidad.

Se puede ahorrar mucha memoria utilizando subrutinas, pero las llamadas a subrutinas consumen tiempo y por lo tanto son indeseables si lo que quieres es una ejecución muy rápida. Como ya has visto, si optas por las subrutinas, tienes que ser muy cuidadoso con la estructura del programa para poder recuperar algo de la velocidad perdida.

Análogamente, `LET A = VAL «100»` está muy bien para ahorrar memoria, pero resulta desastrosa cuando

se compara con `LET A = 100`, que consume más memoria, pero está más indicada para aumentar la velocidad.

VARIABLES

Si quieres acelerar tus programas, te resultará muy útil el conocimiento de cómo se almacenan las variables en la memoria. El área de variables se limpia con `RUN` o `CLEAR`, y las variables se crean a medida que van apareciendo. Generalmente las nuevas variables se añaden extendiendo hacia arriba el área de variables.

Consideremos una situación en la que se crea una variable que es una cadena de caracteres y a continuación una matriz numérica. Después puedes añadir algo a la cadena de caracteres. Si la cadena ya había sido creada antes en el programa, todas las variables creadas con posterioridad tendrán que desplazarse hacia arriba en la memoria, para acomodarse a las condiciones de la cadena.

Echa un vistazo a este ejemplo:

```
100 LET T$=""
110 DIM A(1000)
120 LET T$=T$+"LO QUE SEA"
```

En casi todas las versiones de BASIC se puede ahorrar bastante tiempo permutando las dos primeras líneas de programa, es decir, dimensionando la matriz antes de definir la cadena de caracteres. Tal como está el programa, puede que haya que mover 5000 bytes cada vez que se suma «LO QUE SEA» a T\$.

Hay una regla general, por la que si usas variables en vez de números, ahorrarás gran cantidad de tiempo. Puede que esto no te parezca gran cosa, pero sí hay muchos bucles en los que constantemente se manejan números, el tiempo total ahorrado podría ser considerable. En el caso del Spectrum parece cierto lo contrario. `LET C = 10 + 10` tarda 3 milisegundos, mientras que `LET C = D + D`, donde `D = 10`, tarda 4,2 milisegundos.

ULTIMAS NOVEDADES... ULTIMAS NOVEDADES... ULTIMAS NOVEDADES...

POV

SOFTWARE

NAPOLIS, 98, 1.º 3.ª - T

08013 BARCELONA

EL PODER D

SOLICITANOS TUS PR

LASER ZONE (Spectrum)	975 Pts
GRIDRUNNER (Spectrum)	975 Pts
ASTRO BLASTER (Spectrum)	1.275 Pts
FRENZY (Spectrum)	1.275 Pts
FRIDGE FRENZY (Spectrum)	1.800 Pts
QUINTIC WARRIOR (Commodore)	1.275 Pts
STRONTIUM DOG (Spectrum)	1.800 Pts
STRONTIUM DOG (Commodore)	2.100 Pts

¡¡¡ATENCIÓN!!! POWER, SOFTWARE, medios legales a su alcance, cualquier importación o duplicados ilegales q

BROAD STREET

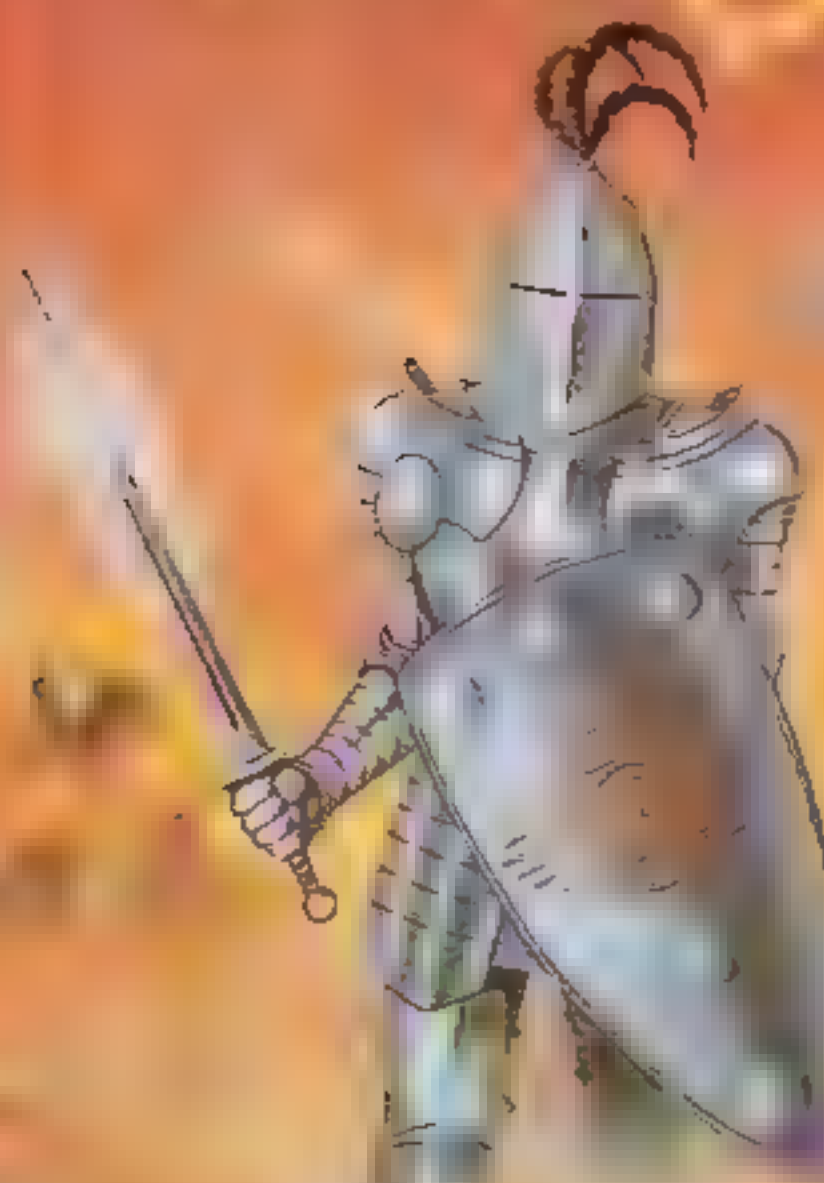
When the music stops, the mystery begins



BROAD STREET

7 Personajes, 10 acordes perdidos, 15 horas, 48 guardias de tráfico, 95 estaciones de metro, 45.000 m.² de Londres, 7 millones de londinenses, 943 pantallas con trepidantes acciones.

EVIL CROWN



EVIL CROWN

Venciendo en los torneos reales podrás llegar a ser Rey. ¿Podrás apoderarte del resto del mundo medieval y adueñarte de la corona malvada?



TURBO
COMMODORE
JOYSTICK
SPECTRUM

STRONTIUM DOG

Johnny Alpha es un agente cazador de recompensas de electrobengalas, que desobligándoles a dejar de d Buena suerte

ULTIMAS NOVEDADES... ULTIMAS NOVEDADES... ULTIMAS NOVEDADES...

ULTIMAS NOVEDADES... ULTIMAS NOVEDADES... ULTIMAS NOVEDADES...

EVER

CENTURY

RE, S.A.

232 24 61 - 232 25 52

ONA (SPAIN)

E LA MENTE

GRAMAS FAVORITOS

PURPLE TURTLES (Commodore)	1.275 Pts
BROAD STREET (Spectrum)	2.400 Pts
BROAD STREET (Commodore)	2.600 Pts
EVIL CROWN (Spectrum)	2.400 Pts
EVIL CROWN (Commodore)	2.600 Pts
TREASURE ISLAND (Spectrum)	2.400 Pts
TREASURE ISLAND (Commodore)	2.600 Pts
ZAKIL WOOD (MSX)	2.800 Pts

WARE, S.A. Perseguirá por todos los tipos de piratería, comercialización, sobre sus programas se practique.



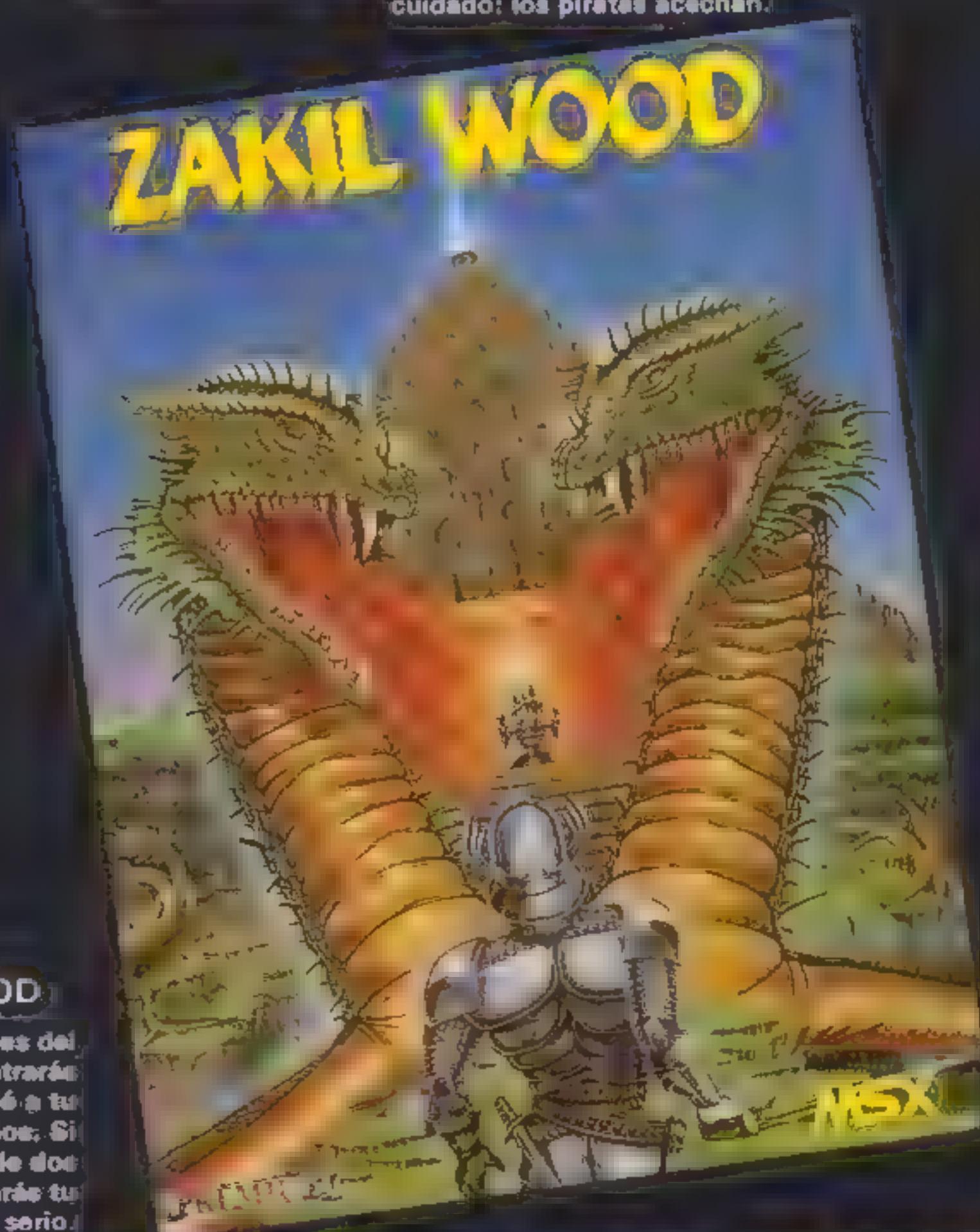
TREASURE ISLAND

Si logras desembarcar de 'La Hispaniola' y vencer a la amotinada tripulación, podrás adentrarte en la laberíntica isla e iniciar la búsqueda del Tesoro. Pero cuidado: los piratas acechan.



ZAKIL WOOD

Investigador destructor, en el futuro Johnny lleva consigo umbrarán a sus enemigos, parará y bombas de tiempo en tu misión.



ZAKIL WOOD

En las profundidades del Bosque de Zakil, encontrarás al rubí que protegió a tu Pueblo durante siglos. Si logras vencer al Pyral de dos cabezas, comenzarás tu aventura en serio.

ULTIMAS NOVEDADES... ULTIMAS NOVEDADES... ULTIMAS NOVEDADES...

P y R

¿Por qué estas reglas de velocidad se aplican solamente a algunas máquinas?

No hay normas rígidas sobre cómo escribir los programas en BASIC para arrancarle el esfuerzo supremo de velocidad, como tampoco las hay sobre la forma de escribir intérpretes; después de todo son programas de máquinas. Pueden contener compromisos parecidos a los de los programas que tú puedas escribir.

La velocidad de un intérprete dependerá de la forma en que esté escrito y de las prestaciones que el fabricante quiera incorporar en su BASIC.

P y R

El BASIC me sigue pareciendo demasiado lento, pero creo que programar en código máquina es muy tedioso. ¿Qué otra cosa puedo hacer?

Hay una especie de término medio, constituido por los lenguajes compilados en vez de interpretados. Lo que más tiempo requiere es ir interpretando el BASIC mientras el programa está corriendo.

El programa compilado utiliza en cambio otro tipo de programa —parecido a los familiares ensambladores de la programación en código máquina— que convierte el programa escrito en un lenguaje de alto nivel en un programa en código máquina, antes de empezar la ejecución. Esto significa que después de la compilación, se ejecuta un programa en código máquina en vez de un programa de alto nivel.

FUNCIONES MATEMATICAS

Utiliza las rutinas de medida de tiempos para comparar estas dos maneras de calcular la misma cosa:

```
200 LET C=4*4*4*4
```

o bien

```
200 LET C=4^4
```

A lo mejor te creías que las funciones que la máquina lleva incorporadas resultan de ejecución más rápida. He aquí otro ejemplo del conflicto entre economía y velocidad de ejecución.

Se ha sugerido a veces líneas como la siguiente:

```
210 IF X>Y THEN LET Y=Y+1
220 IF X<Y THEN LET Y=Y-1
```

pueden ganar mucho en velocidad sustituyéndolos por:

```
210 LET Y=Y+(X>Y)-(X<Y)
```

Efectivamente, el BASIC de Sinclair permite este tipo de comparaciones y la línea única puede parecer una forma más elegante de programación, pero basta hacer una comparación con la rutina de medida de tiempos para ver que es más rápida la versión de dos líneas.

Prueba en ambos casos con diversos valores de X e Y.

MULTIPLICACION Y DIVISION

Las expresiones $C = D * 0.5$ y $C = D / 2$ realizan exactamente el mismo cálculo, pero te encontrarás que la multiplicación es algo más rápida.

Prueba las siguientes sugerencias para ver cuál es más rápida y anota los resultados por si te hacen falta en el futuro.

```
200 LET C=10+10
200 LET C=D+D(donde D=10)
200 LET C=10*10
200 LET C=10/10
```

```
200 LET C=10+PI
200 LET C=SIN 10
200 LET C=COS 10
200 LET C=TAN 10
200 LET C=VAL "10"
200 LET C=10
200 LET C=D(donde D=10)
200 PRINT AT 21,0;"TEST"
200 PRINT AT 21,0;A$(donde A$="TEST")
200 PRINT AT 21,0;10+1000+500+5.5
200 PRINT AT 21,0;D+E+F+G
    (donde D=10,E=1000, etc.)
```




CLASIFICACION Y BUSQUEDA

En un próximo artículo de INPUT nos ocuparemos en profundidad del tema de la clasificación. La ordenación de **Shell-Metzner**, que veremos en dicho artículo, sería la más adecuada para utilizar en cualquier aplicación que implicara la ordenación de, por ejemplo, más de cien elementos. Tiene la particularidad de que cuanto más datos tienes que manejar, más rápida es clasificando. Es desde luego un método mucho más rápido que el cono-

cido método de la burbuja, aunque es también un ejemplo de sacrificio de memoria en aras de una mayor velocidad.

Muchas veces se considera el problema de la búsqueda junto con el de la ordenación. En este caso, en vez de poner en orden una colección de datos, lo que se pretende es recuperar un elemento en particular (o un conjunto de elementos relacionados) con la mayor rapidez posible. Suponte que tienes en tu ordenador una lista de números de teléfono, y que necesitas el

número de Alberto Gómez e Hijos, Papeles Pintados. Naturalmente lo que quieres es que la máquina tarde menos en encontrar el número que si tienes que buscarlo en la guía de teléfonos. Por eso interesan rutinas lo más rápidas posible.

La búsqueda en serie, hace lo mismo que una persona que recorre una lista en un papel, nombre a nombre. El listado es el siguiente:

```
10 DATA "ANTILOPE", "BURRO",  
      "CABRA", "DELFIN",
```


REGLAS PARA AUMENTAR LA VELOCIDAD DEL BASIC

- Utiliza nombres de variables cortos. Siempre que sea posible, una sola letra.

- Sitúa todas las subrutinas de uso frecuente al principio del programa.

- Utiliza bucles FOR ... NEXT con preferencia a bucles del tipo

```
100 LET X = X + 1: IF x < 20
    THEN GO TO 1000.
```

- Utiliza números bajos para las líneas. Empieza a programar a partir de la línea 10 y no a partir de la 1000.

- Emplea las cortas rutinas en código máquina desde el BASIC para el *scrolling*, etc. Conviértete en un ávido coleccionista de las rutinas que se publican en código máquina.

- En los programas de cálculo, de-

fine funciones para hacer los cálculos repetitivos; por ejemplo es mejor poner

```
10 DEF FN A (x) = x - (INT
    (x/360))
```

```
20 LET número = FN A (número)
    que poner
```

```
10 IF número 360 THEN LET
    número = número - 360: GO
    TO 10
```

- Planifica tu programa sobre el papel antes de empezar a teclear. De esta forma evitarás la pérdida de tiempo que implican los excesivos saltos arriba y abajo, indicativos de un programa mal planeado.

- Recuerda que las subrutinas ahorran memoria pero gastan tiempo.

- Elimina todos los denominadores comunes superfluos. Sustituye

```
10 LET X = Y/100 + Z/100 por
10 LET X = (Y + Z)/100
```

- Evita los GOTOs siempre que sea posible

- Siempre que puedas, escribe varias sentencias en una línea.

- Elimina del programa todos los espacios innecesarios, las líneas en blanco y las sentencias REM.

- En las sentencias IF, sitúa primero las condiciones con más probabilidad de ser falsas.

- Utiliza de nuevo los nombres de variables y las variables de bucle en vez de definir unos nuevos.

```
"ELEFANTE","GORRION",
"IGUANA","KOALA","LEON",
"PERRO"
20 RESTORE 10: DIM B$(10,8)
30 CLS : FOR i=1 TO 10: READ
    B$(i): PRINT AT i,5;B$(i)
    : NEXT i
40 POKE 23658,8: INPUT
    "Escribe la palabra ";A$
50 FOR x=1 TO 10
60 IF B$(x, TO LEN A$)=A$
    THEN PRINT FLASH 1;AT x,
    13;"";FLASH 0;"HALLADA":
    GOTO 40
70 NEXT x
```

Aunque la búsqueda serial es una rutina muy conocida, no es muy rápida. La búsqueda binaria o dicotómica no es mucho más difícil de programar, pero es de ejecución mucho más rápida, si bien la diferencia se advierte mejor cuando las listas son largas.

La mayor velocidad de la búsqueda binaria se debe a que, a diferencia de la búsqueda en serie, no tiene que examinar todos los elementos de la lista. Previamente los datos tienen que estar ordenados por orden alfabético o numérico, y el ordenador mira primero al elemento que hay en el centro de

la lista. Desde este punto se mueve hacia arriba o hacia abajo, reduciendo a la mitad cada vez la lista restante, ya que va comparando cada elemento que se encuentra con el que va buscando. Inicialmente no busca una coincidencia perfecta, como en la búsqueda serie, sino que simplemente observa si la primera letra es más alta o más baja que la del elemento deseado.

```
10 CLS: RESTORE
20 LET t=10: LET b=1
30 DIM n$(10,10)
40 FOR c=1 TO 10
50 READ n$(c)
60 NEXT c
70 INPUT"ANIMAL A ENCONTRAR"
    ;a$
75 LET a$=a$+"[10*ESPACIO]"
    (TO 10-LEN a$)
80 PAUSE 50
95 CLS
100 IF n$(t)=a$ THEN PRINT
    n$(t),t: GOTO 200
110 IF n$(b)=a$ THEN PRINT
    n$(b),b: GOTO 200
120 LET p=INT (0.5+(t+b)/2)
130 IF n$(p)=a$ THEN PRINT
    n$(p),p: GOTO 200
```

```
140 IF n$(p)>a$ THEN LET t=p
150 IF n$(p)<a$ THEN LET b=p
160 IF t-b=1 THEN PRINT
    "[3*ESPACIO]NO HALLADO":
    GOTO 200
170 GOTO 100
200 IF INKEY$="" THEN GOTO
    200
210 RUN
580 DATA "ANTILOPE","BURRO",
    "CABRA","DELFIN",
    "ELEFANTE","GORRION",
    "IGUANA","KOALA","LEON",
    "PERRO"
```

Para acelerar los programas de BASIC no hay una respuesta única. Los mejores resultados se obtienen con una atención cuidadosa a muchos detalles aparentemente insignificantes y con mucha experimentación. Como cada programa es un caso especial, nunca puede haber reglas infalibles sobre la forma en que hay que escribir programas. Siempre hay cosas nuevas por descubrir y en parte a ello se debe la fascinación de la programación. Es muy probable que los mayores aumentos de la velocidad de ejecución de tus programas se deban a tus propios descubrimientos.

PROGRAMA TRADUCTOR DE TEXTO

En uno u otro momento todos hemos sentido la tentación de «personalizar» algunos programas en lo que respecta a los textos que aparecen en pantalla.

Unas veces nos gustaría cambiar alguna palabra por otra, otras la inten-

ción pudiera ser introducir rótulos a nuestro antojo, aprovechando espacios en blanco. Quizá para una mayoría sería una gran satisfacción poder disponer de una versión española «hecha por nosotros mismos».

Aunque la idea es tentadora, tiene algunas dificultades su puesta en prác-

tica cuando las zonas de texto se hallan entremezcladas con el resto del código máquina.

La primera dificultad estriba en localizar el texto dentro de la RAM, máxime si se tiene en cuenta que por lo general se halla muy fragmentado.

El siguiente problema nos lo encontramos en la propia traducción o sustitución. Ocurre algo parecido al doblaje de las películas (la frase española deber ser igual de larga y con una vocalización parecida a la del idioma original). En nuestro caso resulta totalmente imprescindible que el nuevo texto sea como máximo de la longitud del original. Si nos «salimos» podemos garantizar que el programa dejará de funcionar correctamente, en su totalidad o en alguna de sus partes.

Nos toca ahora sustituir, carácter a carácter, en la RAM el texto antiguo por el nuevo.

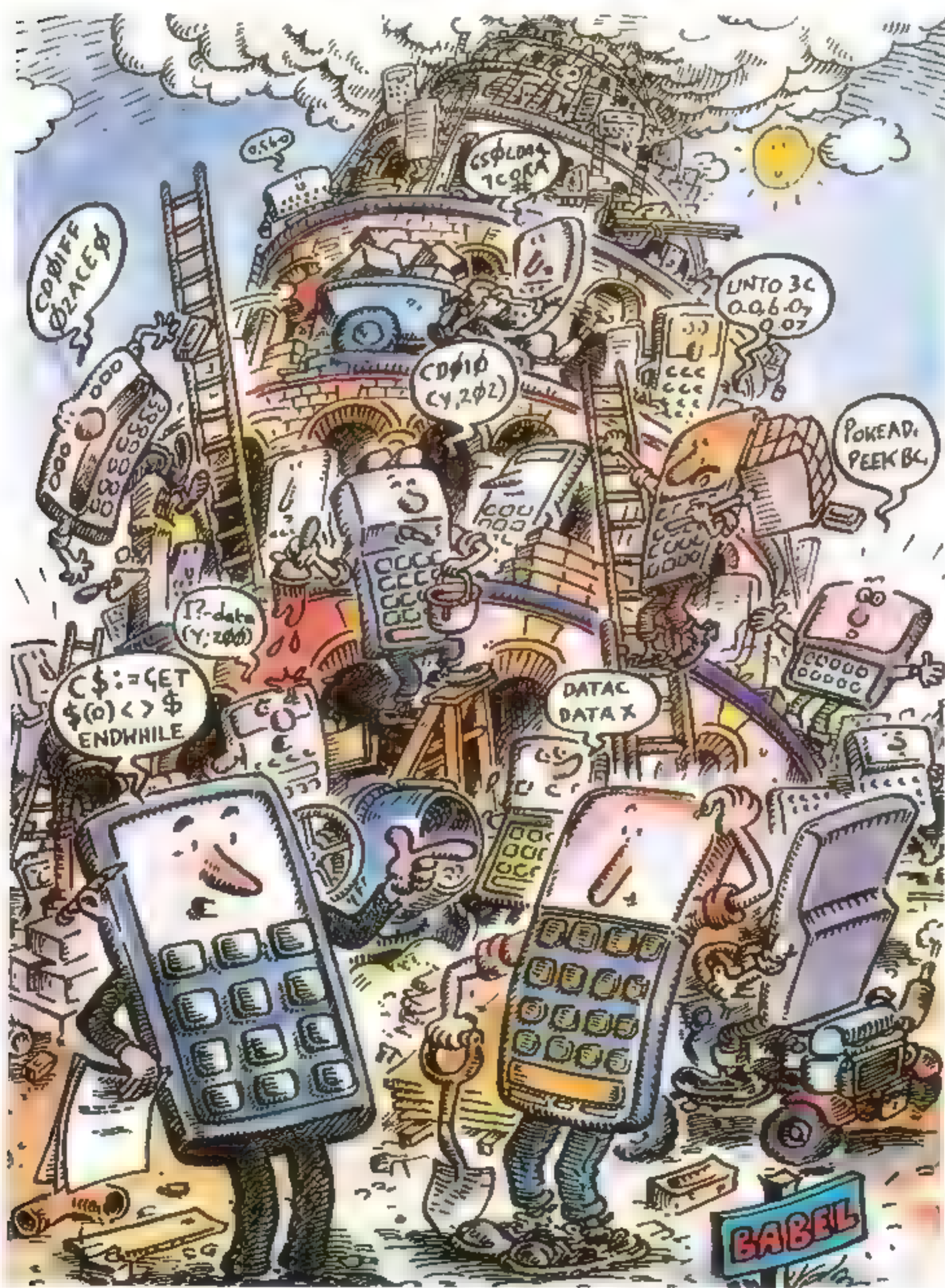
La experiencia nos demostrará que al rodar (RUN) el programa la primera vez (y a veces alguna más ...) observaremos con cierto disgusto que la cosa no ha quedado como deseábamos.

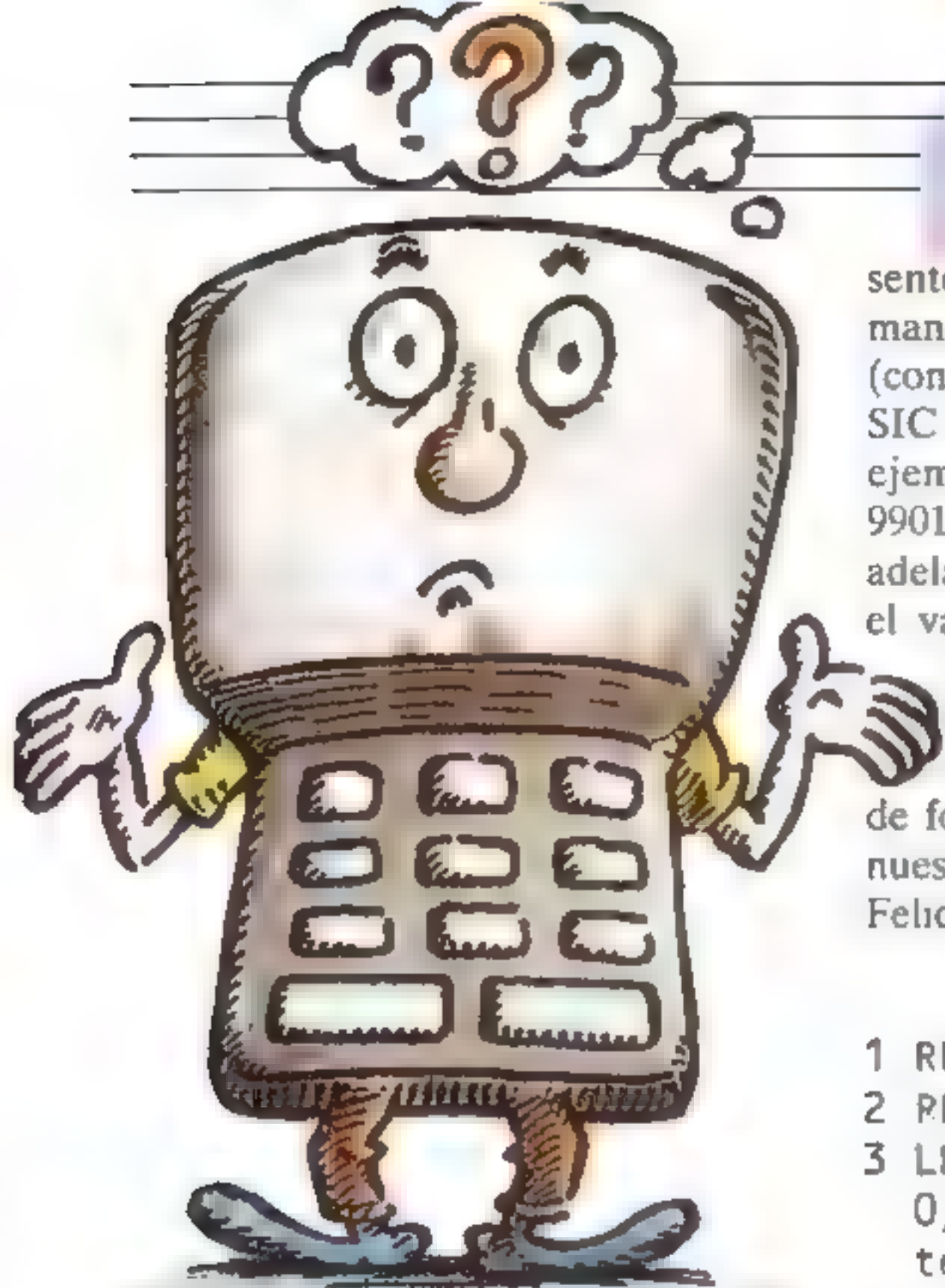
Quizá sea este el momento de probar el programa que os presentamos y que se rompa la cabeza el ordenador.

Su uso es muy sencillo ya que basta con seguir paso a paso las instrucciones que van apareciendo en pantalla y por eso nos centraremos sólo en su estructura y filosofía.

La idea principal estriba en representar sólo aquellos caracteres ASCII que no sean 'de control' (valores 1-31) o 'tokens' (165-255). Los restantes se sustituyen en pantalla por un cuadrado negro (ver instrucción 13) para facilitar la lectura del texto. Con ayuda de **ENTER** podemos desplazarnos sobre el texto para cambiarlo o dejarlo como está.

Una vez finalizada la modificación del texto que deseemos, lo más recomendable es pasarla a cinta (o *diskette*) antes de tratar de ver como nos ha quedado ejecutando el programa completo. (Si el texto es largo esta maniobra debe hacerse de vez en cuando para no perder el trabajo ante un fallo de energía o un *crack* del programa).





ma). El programa incluye también esta facilidad (ver opción 4, instrucciones 20 a 25).

Es muy importante traducir o modificar sólo aquellos caracteres que sepamos con certeza que componen un texto y abstenernos cuando hay duda: puede tratarse de parte de instrucciones en código máquina (ojo a las letras o símbolos sueltos).

Para saltar a explorar otra parte de la RAM distinta a donde nos encontramos sólo debemos teclear 'fin' seguido de **ENTER** y elegir la posición de memoria a partir de la cual queremos trabajar. Si lo deseamos podemos echar un vistazo rápido a una amplia zona de memoria (opción 1) para elegir cómodamente la zona donde queremos centrar la traducción (ver instrucciones 10 a 12).

La práctica nos ha demostrado que resulta muy conveniente no perder idea del conjunto y por eso el programa permite visualizar un renglón 'por delante' del que estamos traduciendo y además va dejando en pantalla los últimos renglones manejados (ver instrucciones 13 a 19).

Las instrucciones 2 y 3 nos sirven de ayuda para cargar en RAM la parte de C/M que deseamos traducir.

Por último indicaremos que el programa es 'reubicable' (ver primera

sentencia de la instrucción 3). De esta manera podemos situarlo fácilmente (con MERGE) dentro de la parte BASIC que controla la parte C/M. Por ejemplo podemos elegir las posiciones 9901 en adelante, en lugar de la 1 en adelante y sólo se requiere modificar el valor de la variable 'pos' a 9901.

Esta forma resulta cómoda para alternar la traducción con el rodaje del programa inicial para ver de forma práctica como va quedando nuestro trabajo (GO TO pos). Felices traducciones.

```
1 REM *** MOLISOFT 85 ***
2 REM
3 LET pos=1: CLS: PRINT AT
  0,0;"Programa traductor de
  texto";AT 1,0;"-----
  -----";AT 2,0;
  "MOLISOFT 85";AT 5,0;" 1-
  Vistazo general";AT 7,0;
  " 2- Traduccion";AT 9,0;
  " 3- Cargar C/M de cassette
  ";AT 11,0;" 4- Archivar C/M
  en cassette";AT 15,3;
  INVERSE 1;"Pulse opcion"
4 LET a$=INKEY$: IF a$=""
  THEN GO TO pos+3
6 LET n=CODE a$: LET n=n-48:
  IF n<1 OR n>4 THEN GO TO
  pos+2
7 CLS : IF n=1 OR n=2 THEN
  INPUT " posicion de memoria
  ? ";k: GO TO ((n=1)*(pos+9)
  +(n=2)*(pos+12))
8 GO TO ((n=3)*(pos+29)+(n=4)
  *(pos+19))
10 PRINT " INVERSE 1;"
  posiciones de ":";k;" a ";
  k+99; INVERSE 0: PRINT :
  FOR i=k TO k+99: LET c=
  PEEK i: LET c=((c<32)*143
  +(c>=32)*c): PRINT CHR$ c
  ;: NEXT i
11 INPUT "seguir (ENTER)/ NO
  seguir(fin + ENTER)";p$:
  IF p$<>"" THEN GO TO 3
12 LET k=i: GO TO 10
13 PRINT " INVERSE 1;
  "posiciones de ":";k;" a "
  ;k+63; INVERSE 0: PRINT :
  FOR i=k TO k+63: LET c=
  PEEK i: LET c=143*(c<32
```

```
OR c>164)+c*(c>=32 AND c<
=164): PRINT CHR$ c;:
NEXT i
14 PRINT " INVERSE 1;
  "traduccion de ":";k;" a "
  ;k+31; INVERSE 0: PRINT :
  FOR i=k TO k+31: LET c=
  PEEK i: LET t$="": IF c>=
  32 AND c<=164 THEN INPUT
  ""fin"" o traducc. de "
  +CHR$ 18+CHR$ 1+CHR$ c+
  CHR$ 18+CHR$ 0;" "; t$
15 IF t$="fin" OR t$="FIN"
  THEN LET i=k+63: NEXT i:
  GO TO pos+2
16 IF t$<>"" THEN POKE i,CODE
  t$
17 LET c=PEEK i: LET c=143*
  (c<32 OR c>164)+c*(c>=32
  AND c<=164): PRINT CHR$ c;
18 NEXT i: BEEP 2,10: INPUT
  "repetir ? s/n ";p$: IF
  p$="S" OR p$="s" THEN GO
  TO pos+13
19 LET k=k+32: FOR h=1 TO 32:
  PRINT "=";: NEXT h: PRINT
  : GO TO pos+12
20 CLS : INPUT "nombre ? ";n$
  : INPUT "posic. mem. inic.
  ? ";p1: INPUT "longitud en
  bytes ? ";long: PRINT AT
  10,0;"PONGA EN MARCHA LA
  CASSETTE PARA";AT 12,0;"
  GRABAR EL NUEVO C/M";AT
  14,0;"PULSE ENTER";AT
  16,0;"(no olvide
  desconectar EAR)": PAUSE 0
21 IF n$="" THEN LET n$=" "
22 SAVE n$CODE p1,long
23 CLS : PRINT AT 10,0;"
  REBOBINE LA CASSETTE PARA"
  ;AT 12,0;"VERIFICAR LA
  GRABACION";AT 14,0;"PULSE
  ENTER";AT 16,0;"(no olvide
  conectar EAR)": PAUSE 0
24 VERIFY n$CODE
25 STOP
30 INPUT "nombre?( "" si no
  desea )";n$: CLS : PRINT
  AT 10,0;"PULSE ENTER";AT
  12,0;"PONGA EN MARCHA LA
  CASSETTE PARA";AT 14,0;"
  CARGAR EL C/M": PAUSE 0
31 LOAD n$CODE
32 STOP
```


NUEVOS PERIFERICOS **MHT**

CON SONIDO POR TV

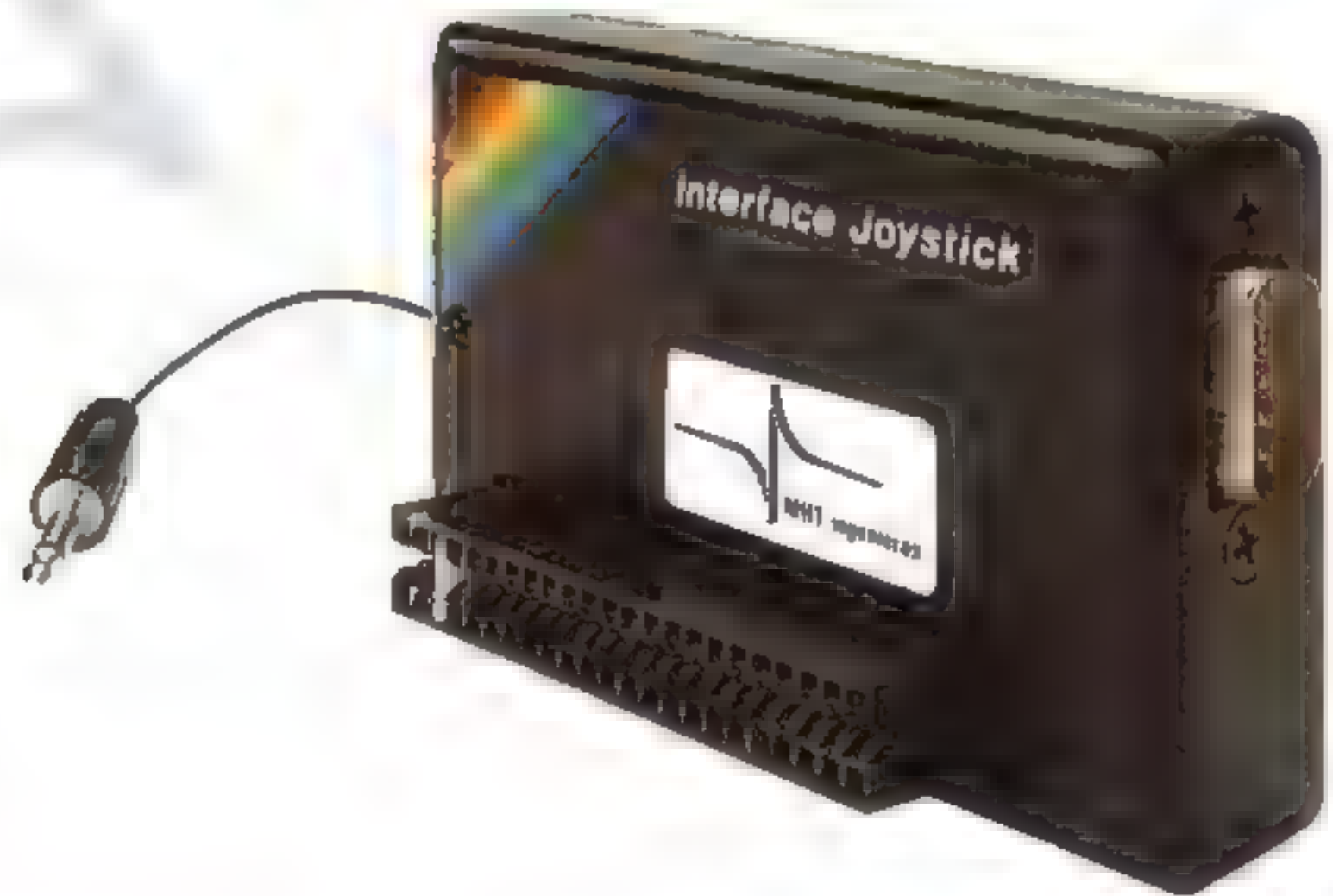
INTERFACE MULTIJOYSTICK

Viene preparado para que juegues tu sólo en opción Kempston, cursores o sinclair o bien con tu amigo en las opciones Sinclair-1 y Sinclair-2 para dos jugadores. Y todo ello con el sonido amplificado a través del altavoz de tu TV.



INTERFACE TIPO KEMPSTON

Aparte de poder manejar tu Joystick con juegos preparados para la opción Kempston, podrás escuchar igualmente su sonido amplificado a través del altavoz de tu TV.



*CARGA EL PROGRAMA, TECLEA, JUEGA... y ESCUCHALO POR TV. con los NUEVOS PERIFERICOS **MHT***

Distribuido por



S.A.

Sánchez Pacheco, 78
28002 Madrid
Teléfono 413 92 68

DE VENTA EN TIENDAS ESPECIALIZADAS.

SERVICIO POST-VENTA GARANTIZADO

ES UN PRODUCTO DESARROLLADO Y FABRICADO
EN ESPAÑA POR **MHT INGENIEROS**

EL RATON, UN SIMPATICO ACCESORIO

El reto aparece casi al mismo tiempo en que comienza a desarrollarse la ciencia informática. La comunicación hombre/ordenador siempre ha sido de naturaleza artificiosa. Desde luego, un teclado no es la forma más amigable, aunque pueda ser rápida, de acceder a un ordenador.

Sin embargo un teclado representa un enorme avance con respecto a las tarjetas y cintas perforadas que se empleaban hace una década con el mismo fin. La interactividad estaba vetada a prácticamente cualquier usuario. La divulgación de los lenguajes de alto nivel interpretados (por contra a compilados) supone un enorme paso adelante en las comunicaciones hombre/ordenador. Se puede escribir una sentencia y conocer rápidamente si es comprendida y cómo por el sistema.

Los programas diseñados para cubrir una aplicación específica se ponen al alcance de cualquiera que precise los servicios que puede ofrecer un ordenador, sin necesidad de conocer una palabra sobre programación. En este gran apartado se pueden incluir también los videojuegos en cinta o *diskette*. Es particularmente en esta categoría donde el teclado se muestra antipático, sobre todo en los juegos de acción (en los que más cuentan los reflejos) que precisan mayor interactividad capaz de aceptar respuestas inmediatas. Hasta ahora, el periférico más divulgado que cuenta con las características de facilidad de manejo y «amigabilidad» con el usuario es el *joystick*.

No obstante, para determinados juegos y aplicaciones el *joystick* sigue presentando carencias y a veces resulta incluso incómodo. Cabe resaltar que una gran mayoría de modelos nos obligan a utilizar ambas manos para su control «una ha de agarrarlo con fuerza por la base».

Es en plena meca de la microelectrónica e informática, en California,

donde el Instituto de Investigación de Stanford (SRI) comenzó a desarrollar un nuevo dispositivo más identificable con el usuario, a finales de los sesenta. Una de las soluciones consistió en una diminuta cajita capaz de ajustarse en la concavidad formada por la palma de la mano. En la parte anterior asomaban dos pulsadores, semejando a un par de ojos, y también disponía de un cable de conexión al ordenador que recordaba a un rabo. Esta unión de apariencias es la que llevó a bautizar al dispositivo como «ratón».

Hasta hace bien poco el ratón no era un accesorio demasiado utilizado con los microordenadores. Fué el Lisa de Apple Computer quien dió este primer paso con intención de popularizarlo y posteriormente lo hizo compañero imprescindible del McKintosh, una versión de menor precio del anterior. En los últimos tiempos se ha venido utilizando cada vez más, pero es realmente ahora cuando comienza a llegarle la oportunidad a ordenadores de precios más modestos.

En esencia, un ratón se comporta como un *joystick* por lo que al ordenador afecta. Inicialmente fué concebido para ser un dispositivo que se desplaza por una superficie plana —una mesa— en cualquiera de los sentidos y direcciones, bajo el control directo de una mano. Mediante *software* se conseguiría que el cursor se desplace por toda la pantalla siguiendo el mismo movimiento aplicado del ratón. Esto es obviamente mucho más rápido que el empleo de las teclas de cursor para hacer lo mismo.

Los dos pulsadores, que se encuentran siempre al alcance de algún dedo, actúan de modo similar a los de «fuego» que incorporan los *joysticks*.

Un funcionamiento de este tipo permite que un usuario no experimentado pueda sacar rendimiento inmediato a programas de aplicación maneja-

VENTANA POR LA QUE ASOMA UNA PARTE DE LA ESFERA. EN TAL MODO QUE PUEDE REALIZAR EL MOVIMIENTO DE RODADURA POR FRICCIÓN CON LA SUPERFICIE PLANA. AL SER DE MENOR DIAMETRO QUE LA BOLA, SIRVE PARA RETENERLA EN EL INTERIOR DEL RATON.

CUATRO PEQUEÑAS BOLAS METALICAS SE ENCARGAN DE ELEVAR LA BASE DE LA CAJA POR ENCIMA DE LA SUPERFICIE DE TRABAJO, FACILITANDO EL DESPLAZAMIENTO DEL RATON.

ESFERA DE GOMA MACIZA. EN OTROS MODELOS SE USAN DIVERSOS MATERIALES COMO EL PLASTICO O EL ALUMINIO. SIEMPRE DEBE SER PESADA, PARA EVITAR QUE HAYA DESLIZAMIENTO Y NO RODADURA.

DISCO CON BANDAS REFLECTANTES, UNIDO SOLIDARIAMENTE AL EJE DE UN RODILLO.

MICROINTERRUPTORES. ACTUAN COMO LOS PULSADORES DE DISPARO EN LOS JOYSTICKS.

VENTANA DE DIAMETRO LIGERAMENTE SUPERIOR A LA BOLA QUE PUEDE GIRAR LIBREMENTE EN SU INTERIOR.

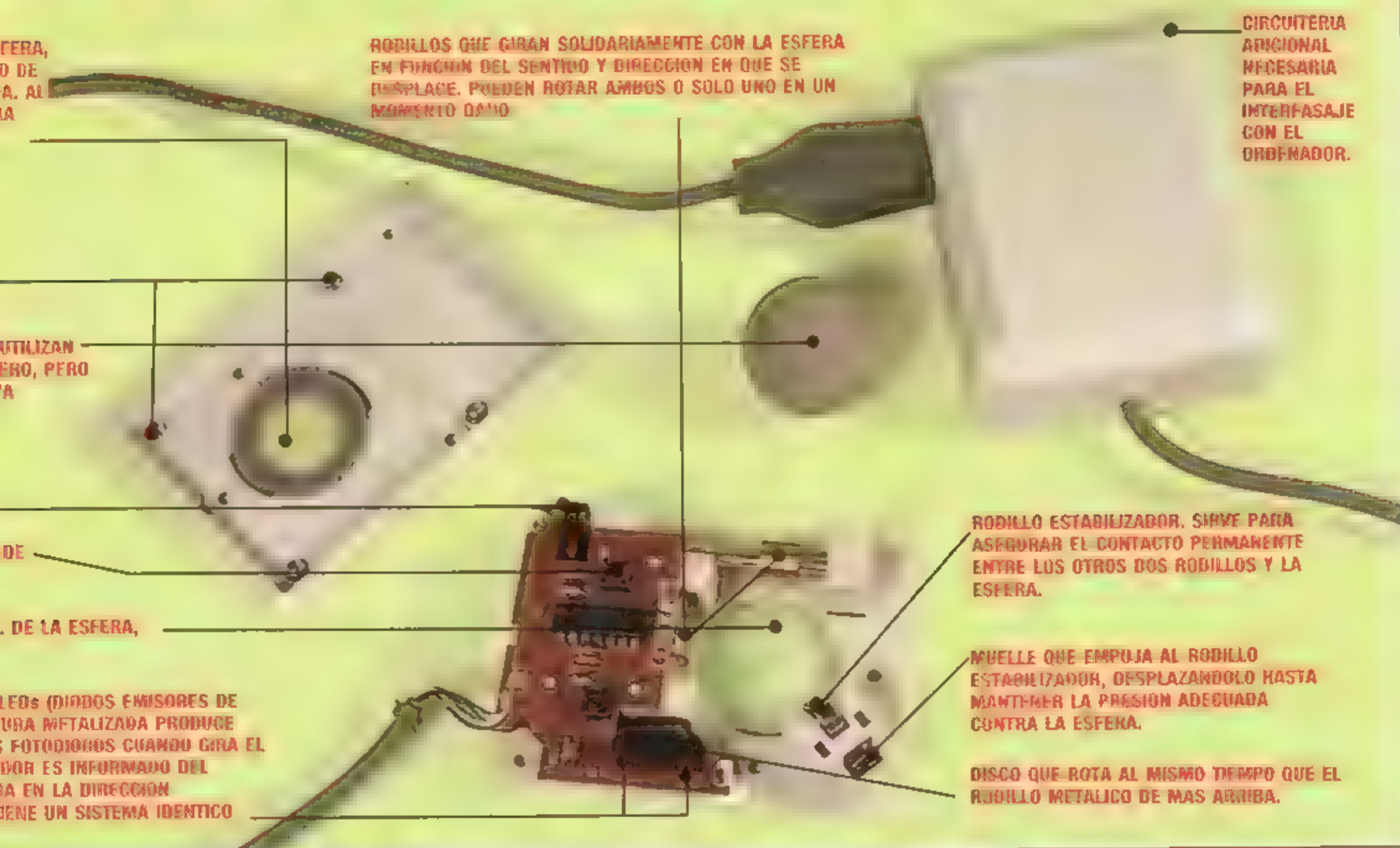
CONJUNTO FORMADO POR DOS FOTODIODOS Y DOS LEDS. LA LUZ REFLEJADA POR LAS BANDAS DE PINTA DE LOS IMPULSOS ELECTRICOS EN LOS TERMINALES DE LA BOLA. UNA VEZ PROCESADA LA SEÑAL, EL QUE INDICA EL SENTIDO Y MAGNITUD DE MOVIMIENTO DE LA ESFERA PERPENDICULAR A ESTE RODILLO. EL OTRO DISCO INDICA EL MOVIMIENTO PARALELO A ESTE RODILLO.

bles a partir de menús. El cursor tomaría la apariencia de una flecha, por ejemplo, en lugar del clásico cuadrado. Así, moviendo la flecha hasta señalar la opción elegida, no hay más que presionar el botón de disparo y continuar eligiendo opciones, introduciendo datos, etc.

Se podría argumentar que esto también se puede conseguir con el *joystick* y en efecto es así, pero a la larga resulta más incómodo y menos «serio» cuando no se trata de programas de juego. Incluso con dicho tipo de programas se ejerce más cómodamente el control, porque los movimientos intuitivos se reflejan de modo más inmediato.

El fundamento del ratón es bien sencillo, una esfera maciza puede rodar libremente cuando asoma por una ventana circular situada en la base de la caja. En sus movimientos de giro obliga a rotar a dos cilindros situados perpendicularmente entre sí. Ambos rodillos llevan en un extremo una rue-

EL RATON POR DENTRO



da ranurada que gira solidariamente con ellos. De esta manera cada una dará vueltas o no, dependiendo de la dirección en la que sea desplazada la caja. Además cada una podrá hacerlo a velocidades de giro diferentes, según la resultante del movimiento del ratón en un momento dado.

Los discos pueden no ser ranurados, como en el caso particular del modelo que estamos describiendo. En

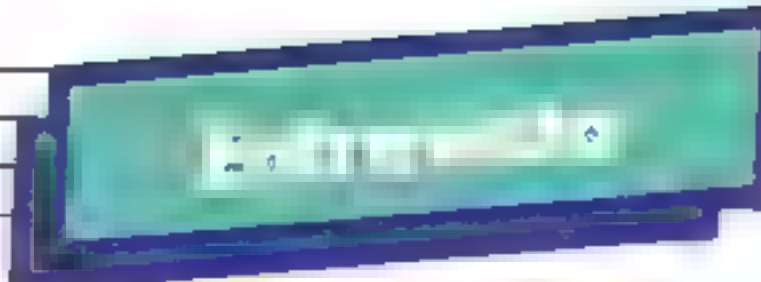
lugar de ventanitas, dispone de pintura reflectante distribuida a intervalos regulares. En una posición fija muy próxima a los discos, se han dispuesto dos fotodiodos y otros tantos LEDs (diodos emisores de luz), de tal forma que la luz emitida por los diodos se refleja en la pintura, encargándose los fotodiodos de detectar en que momento es máxima o mínima la cantidad de luz reflejada que les alcanza, tradu-

ciendolo a impulsos eléctricos. De esta manera y complementados por nueva circuitería electrónica es posible saber en qué dirección y cuánto se ha desplazado la esfera.

Procesada esta información, el ordenador recibe el mismo tipo de datos que le proporcionaría un *joystick*.

En el futuro el ratón dejará de ser únicamente un bicho capaz de provocar terror a cualquier secretaria.

Si se te hace difícil encontrar INPUT
en tu kiosco habitual,
resérvalo por adelantado, o háznoslo saber
para que podamos remediarlo.



MADE IN SPAIN A LA CONQUISTA DE EUROPA

Alguna vez se ha dicho que con la imaginación que tenemos los españoles no tardaríamos en destacar en una de las áreas más importantes de la informática: la creación del *software*. Sin embargo hasta la fecha, la inmensa mayoría de los programas comerciales son de importación, y sólo algunos equipos de jóvenes entusiastas trabajan, principalmente sirviéndose del Spectrum, para colocar sus creaciones en las listas de los programas más vendidos.

Made in Spain es un joven equipo aunque ya veterano, pues si la media de edad de sus componentes ronda los 20 años, cuentan ya con la experiencia de haber vendido 20.000 copias de su primer programa comercial —el **Fred**— en el Reino Unido. Estos jóvenes están demostrando, entre otras cosas, que lo de vender en Europa no es cuestión de medios tecnológicos ni de grandes inversiones, sino más bien de imaginación.

Cuando nos enteramos de que acababan de firmar un contrato con **Mikro-Gen** para la comercialización de su segundo programa, al que han bautizado como **Sir Fred**, nos pusimos en contacto con ellos para que nos hablaran del cómo y del por qué de su trabajo y nos enseñaran, de paso, algo sobre el secreto de su éxito. Para charlar con ellos, nos fuimos a su cuartel general, una habitación en la academia **Mister Chip** en la que tienen todo su equipo.

MADE IN SPAIN

Carlos Granados, Paco Menéndez, Camilo Cela y Fernando Rada son cuatro amigos y compañeros de colegio que cuando cursaban BUP empezaron, medio en serio medio en broma, con esto de la programación. Hoy es-

tán en tercero de carrera, todos en Físicas excepto Paco que hace Telecomunicaciones y básicamente, siguen siendo lo mismo: un grupo de amigos que se reúnen en los ratos libres para hacer programas.

El hecho de haber vendido con completo éxito el **Fred** no cambió su forma de trabajo, y ni siquiera aumentaron sus medios técnicos. Aunque en realidad, el **Fred** supuso más un éxito de popularidad que un éxito económico, por lo menos para ellos, que percibieron un total de 1.700.000 pesetas incluyendo los derechos por la adaptación a otros ordenadores, y que ni siquiera han llegado a saber cuantas copias se vendieron en España.

Por esto último, para la comercialización de su nuevo programa han cambiado su estrategia. «En cuanto estuvo terminado, hicimos un vídeo y se lo mandamos a las principales casas de *software* inglesas, a la vez que les pedíamos que nos contestaran con sus ofertas. Nos respondieron casi todas y entonces decidimos irnos a Inglaterra por nuestros medios para negociar directamente con las cuatro que nos habían hecho las mejores ofertas».

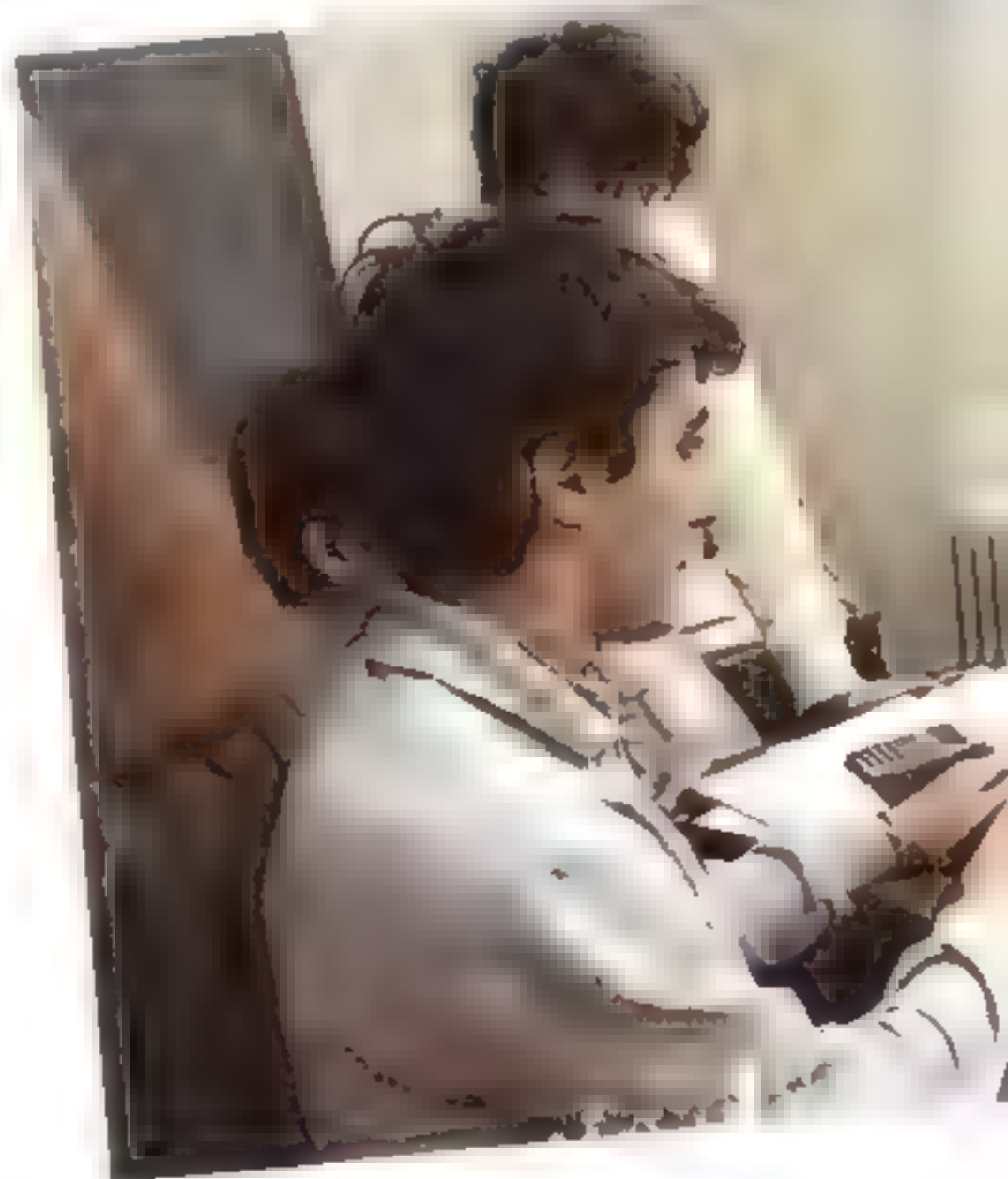
De este viaje surgió un magnífico contrato con **Mikro-Gen**, en el que esta casa se compromete a pagar una libra por copia realizada, garantizando un mínimo inicial de 20.000 copias y estimando el total posible en 50.000 ó 60.000. Aparte de esto negociarán por separado la adaptación del programa a otras máquinas así como la comercialización en otros países excepto España, que se han reservado los autores.

TRABAJANDO COMO SIEMPRE

Como decíamos, sus éxitos poco han cambiado sus costumbres. «Antes nos reuníamos en la casa de uno de

nosotros, hasta que se cansaban y nos íbamos a otra. Ahora trabajamos en esta habitación que nos han dejado en la academia».

Los medios técnicos siguen siendo básicamente los mismos: un par de **Spectrum** conexiados de tal forma que uno actúa como unidad central en la que se teclean las nuevas subrutinas mientras que el otro contiene el resto del programa.



El almacenamiento se efectúa mediante unidades de disco. El resto del equipo lo componen dos viejos televisores, y a raíz de su viaje al Reino Unido, un **Amstrad** en el que ellos mismos realizan la adaptación del programa para esta máquina.

El planteamiento y la división del trabajo son también bastante informales. En realidad el programa se va desarrollando poco a poco a partir de una idea inicial, que en el caso de **Sir Fred** ha sido el movimiento del personaje. «Luego cada uno plantea sus ideas y trabaja en las subrutinas hasta que funcionan. Después de probadas



y perfeccionadas se incorporan al programa y así, una idea tras otra hasta que nos quedamos sin memoria».

PIRATAS Y PIRATILLAS

Sir Fred ha sido convenientemente dotado de protecciones para dificultar el que sea copiado ilegalmente. Sin embargo sus autores no se hacen demasiadas ilusiones al respecto. «Ac-



tualmente, en dos o tres días abren y desprotegen cualquier programa». El problema de la piratería les preocupa relativamente. Lo consideran como un mal difícilmente evitable habida cuenta de nuestra mentalidad meridional (comentan que con los italianos pasa otro tanto). Pero del mismo modo que perdonan a aquellos que en un momento determinado pueden realizar una copia para un amigo, no piensan lo mismo de los piratas industriales, que no sólo copian el programa, sino que además cambian el título y un par de cosas más y luego lo venden como propio.

Sin embargo tienen algunas ideas propias sobre como afrontar este tema. «Una de las causas de la piratería es que los programas suelen ser bastante caros. Si el precio de una cinta fuera menor, 500 pesetas por ejemplo, probablemente no merecería la pena copiarlo. Claro que esto traería consigo dos problemas. De un lado, los distribuidores tendrían que vender muchísimos más para que aceptasen tales precios, y por otro lado habría que convencer a los usuarios de que un programa de 500 pesetas no tiene por qué ser necesariamente malo».

Sea como fuere, y pese a las 800 pesetas por copia que cobran por la distribución de su programa en España, consideran que lo que puedan recaudar aquí, siempre será una pequeña parte de lo que obtendrán de la venta en el resto de Europa.

PROGRAMANDO DE CARA AL FUTURO

Parece fuera de duda que en el peor de los casos **Made in Spain** va a conseguir algunos millones de pesetas con la facturación de su programa. Teniendo en cuenta que la principal inversión que han realizado estos jóvenes es la de su tiempo, una vez cubiertos algunos gastos, entre ellos la publicidad y la fabricación de la versión española, la mayor parte de esta facturación se puede considerar como beneficios. Pues bien, esto no parece tenerles muy impresionados, y, hoy por hoy, no cuentan con formar una em-

presa «seria» para dedicarse plenamente al desarrollo y comercialización de *software*. Tan sólo comentaron que «eso estaría bien porque así tardaríamos cuatro meses en hacer un juego en lugar de todo el año. Así se podría ganar más dinero, pero no lo haremos porque la avaricia rompe el saco»... Consecuentemente piensan seguir alternando sus estudios con la programación.

Con respecto al futuro de la programación en nuestro país nos comenta-

ron que «puede dar mucho dinero. Además no es tan difícil como muchos piensan. Desde luego, es fundamental conocer el código máquina, pero esto ahora es factible dado que la bibliografía sobre el tema es ya abundante, no como cuando nosotros empezamos, pues entonces no había prácticamente nada escrito. También es verdad que los que empiezan ahora tienen el inconveniente de que deben ponerse al nivel actual, que ya es alto». Consideran que merece la pena por-

que «el mercado español puede crecer si se hace algo contra la piratería».

Como quiera que sea, de momento parecen ser muy pocos los que se han dado cuenta de estas posibilidades, aunque ya se oye hablar de algunos equipos que están surgiendo y a los que esperamos conocer pronto por sus éxitos.

De momento hemos dejado a Fernando, Camilo, Paco y Carlos enfrascados en terminar su versión para Amstrad del Sir Fred. Después, en cuanto pasen los exámenes de Enero y Febrero, volverán a los teclados para crear un nuevo programa: «aprovecharemos todo lo que hemos aprendido con Sir Fred para hacer algo completamente distinto, aunque todavía no nos hemos detenido a pensar el qué».

Estamos convencidos de que antes de un año oiremos hablar de un nuevo programa fabricado en España que ocupará los primeros lugares en las listas europeas.

NO OLVIDES EL TELEFONO...



Cuando, por cualquier motivo, nos escribas, no olvides indicar tu número de teléfono. Así nos será más fácil y rápido ponernos en contacto contigo.
Gracias.



Libros para Spectrum

ZX SPECTRUM QUE ES PARA QUE SIRVE Y COMO SE USA
por Tim Langdell
El medio de llevar el Spectrum al límite y más allá
PVP 1.100 ptas.

COMO CREAR TUS JUEGOS SPECTRUM
por R. Rovra
Sea inventor y sorprende a sus contrarios
PVP 7.500 ptas.

18 JUEGOS DINAMICOS PARA TU SPECTRUM
por P. Monsau
La informática se aprende jugando
PVP 650 ptas

PRONTUARIO DEL SPECTRUM
Prontuario Spectrum. Todo lo que hay que saber al alcance de la mano
PVP 350 ptas.

EL SPECTRUM Y LOS NIÑOS,
por Meyer Solomon
Los ordenadores al alcance de los niños. De utilidad a partir de los 7 años
PVP 490 ptas.

ZX SPECTRUM APLICACIONES PRACTICAS PARA LA CASA Y LOS PEQUEÑOS NEGOCIOS
por Chris Callender
Para emplear el Spectrum en algo positivo
PVP 870 ptas.

PROFUNDIZANDO EN EL ZX SPECTRUM
por Diwyn Jones
Para profundizar en los trucos y técnicas
PVP 1.300 ptas.

MICROORDENADORES Y CASSETTES,
por Mike Salem
No p erda más programas, se acabaron los problemas de carga
PVP 800 ptas.

DICCIONARIO MICROINFORMATICO
por R. Tapias
El léxico informático explicado. Contiene anexo de inglés-Español
PVP 990 ptas.

EDITORIAL NORAY, S.A.

San Gervasio de Cassolas 79 - 08022 Barcelona (ESPANA) - Tel (93) 211 11 46

Pedidos a NORAY, S.A.		Libro	Precio	TOTAL
San Gervasio de Cassolas. 79 - 08022 Barcelona				
Nombre				
Apellidos				
Direccion				
Poblacion				
D.P. Telefono				
PRECIO TOTAL PESETAS				

PROGRAMACION DE JUEGOS

circular en cada habitación, porque puede ser que haya puertas que quieras que sólo puedan cruzarse en una dirección, acompañadas de un mensaje tal como éste:

LA PUERTA SE CIERRA DE GOLPE JUSTO DETRAS DE TI

Las líneas de trazos que hay junto al cuarto oscuro indican que el aventurero sólo podrá ir en esa dirección si se cumplen ciertas condiciones. En este caso, la condición es que el aventurero tenga la lámpara y la haya encendido para poder ver las salidas.

Es muy difícil predecir cuántos lugares se pueden meter en una cantidad dada de RAM. La dificultad surge porque hay muchas cosas que se disputan el espacio de memoria en el programa de la aventura: descripciones de lugares, palabras que quieres

que el programa reconozca, número de objetos y lo que quieres que se haga con ellos, número de enigmas y su complejidad, etc.

Cuando hayas escrito unas cuantas aventuras pequeñas y comprobado cuánta memoria ocupan, tendrás una idea de lo que puedes meter en tu máquina.

Los poseedores de un ordenador de 16 K pronto descubrirán que es casi imposible escribir una aventura en gran escala en una cantidad tan pequeña de RAM. Sin embargo, la aventura que desarrollaremos en los siguientes capítulos sólo tiene unos pocos lugares—12 en total—por lo que es suficientemente pequeña para no causarte dolores de cabeza.

El mapa para la búsqueda de la joya del globo ocular podría ser algo parecido a la figura 1. En esta fase las descripciones de los lugares han de man-

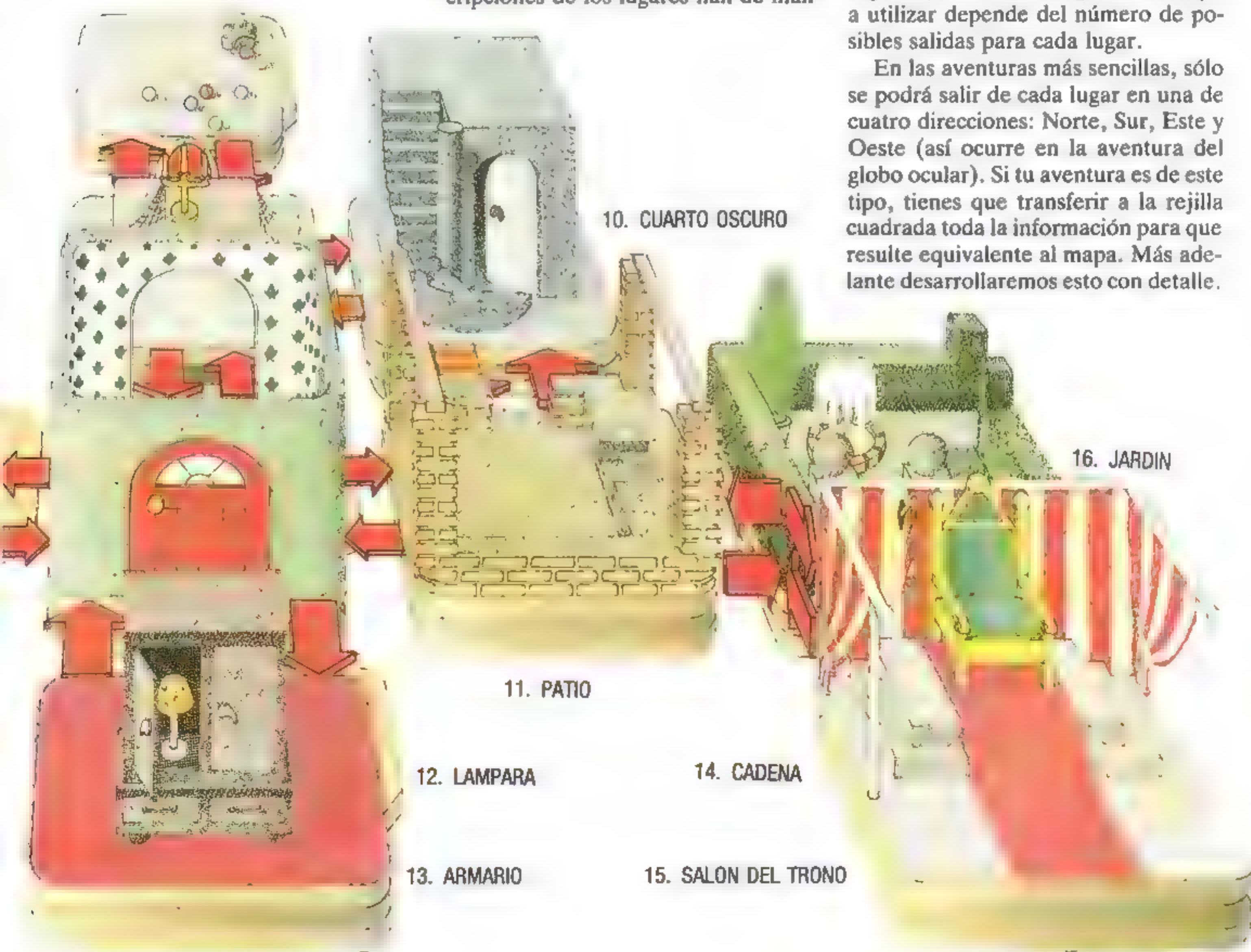
tenerse cortas. Se han dibujado los enlaces y se ha decidido el punto de partida. Esto es muy importante, ya que afecta a la manera en que se aborda la aventura, al orden en que aparecen los objetos y se plantean los *puzzles*.

También están marcados los objetos en sus lugares. Los objetos que aparecerán más tarde, por ejemplo la joya, es mejor listarlos a un lado del mapa.

PASO DEL MAPA A LA RETICULA

Cuando está completo el mapa, la información que contiene se puede transferir a una retícula. Hay dos tipos de retículas que se suelen usar al proyectar juegos de aventuras: una de ellas está basada en cuadrados, como en la figura 2 y la otra basada en octógonos, como en la figura 3. El tipo a utilizar depende del número de posibles salidas para cada lugar.

En las aventuras más sencillas, sólo se podrá salir de cada lugar en una de cuatro direcciones: Norte, Sur, Este y Oeste (así ocurre en la aventura del globo ocular). Si tu aventura es de este tipo, tienes que transferir a la rejilla cuadrada toda la información para que resulte equivalente al mapa. Más adelante desarrollaremos esto con detalle.



Si has utilizado salidas que incluyen Nordeste, Noroeste, Sureste y Suroeste, tienes que emplear la retícula octogonal, aunque este tipo de sistema de rejillas resulta muy complicado.

La última variante de la rejilla se produce si has decidido también subir y bajar. En este caso la mejor solución es utilizar una rejilla separada para cada «nivel» de aventura.

La aventura del ojo precioso está basada sobre una retícula cuadrada, permitiendo sólo salidas por el Norte, Sur, Este y Oeste. A menos que haya una necesidad real de otras direcciones, este tipo de aventuras resulta muy adecuado y hay una manera de obviar las subidas y bajadas. Si utilizas una descripción con frases como

HAY UNA ESCALERA QUE BAJA HACIA EL OESTE

puedes usar la respuesta normal del Oeste o la rutina adecuada.

LA RETICULA

Esta aventura requiere una retícula cuadrada de 6×4 como puedes comprobar contando el número máximo de lugares de tu mapa, de arriba a abajo y de derecha a izquierda. Antes de empezar a trasladar todos los detalles a la retícula, asegúrate de que has numerado todos los cuadrados. Empieza con el número 1 en la parte superior izquierda, y prosigue hasta llegar a la parte inferior derecha. Cuando hayas numerado todos los cuadrados y transferido todos los detalles, la retícula será como la de la figura 4.

EMPIEZA EL PROGRAMA

Ahora tienes una línea histórica y una retícula completa, puedes empezar con el programa.

El primer paso es teclear las descripciones de los lugares, basándote en tu retícula. Tienes que decidir cómo van a ser de largas. Intenta sugerir lo más posible el ambiente de la aventura sin derrochar memoria. Aparte de las descripciones de los lugares, hay que decirle al ordenador en qué direcciones se encuentran las salidas.

Aquí tienes al fin las primeras secciones del programa. Los números de líneas tan altos te permitirán disponer de suficiente espacio en las anteriores secciones de programa a medida que vayas desarrollando el juego.

Teclea esta sección y almacénala (SAVE) en cinta:

```

5000 REM **DESCRIPCION DEL
      LUGAR**
5010 REM **LUGAR 4**
5020 PRINT "Estas fuera de
      un gran edificio"
5030 LET N=0: LET E=0: LET
      S=1: LET O=0: RETURN
5040 REM **LUGAR 7**
5050 PRINT "Estas junto a un
      rio impetuoso"
5060 LET N=0: LET E=1: LET
      S=0: LET O=0: RETURN
5070 REM **LUGAR 8**
5080 PRINT "Estas en un
      bosque petrificado"
5090 LET N=0: LET E=0: LET
      S=1: LET O=1: RETURN
5100 REM **LUGAR 10**
5110 PRINT "Estas en una
      habitacion polvorienta"
5120 LET N=1: LET E=1: LET
      S=1: LET O=0: RETURN
5130 REM **LUGAR 11**
5140 PRINT "Estas en una
      habitacion oscura"
5150 IF LA <> 1 THEN LET N=0:
      LET E=0: LET S=0: LET
      O=0: PRINT "Esta
      demasiado oscuro para
      ver la salida": LET DA
      =1: RETURN
5160 LET N=0: LET E=0: LET
      S=1: LET O=1: RETURN
5170 REM **LUGAR 14**
5180 PRINT "Estas en un
      camino fangoso"
5190 LET N=1: LET E=1: LET
      S=0: LET O=0: RETURN
5200 REM **LUGAR 15**
5210 PRINT "Estas junto a la
      puerta de la ciudad
      secreta"
5220 LET N=0: LET E=1: LET
      S=0: LET O=1: RETURN
5230 REM **LUGAR 16**
5240 PRINT "Estas en la
  
```

```

      entrada principal"
5250 LET N=1: LET E=1: LET
      S=1: LET O=1: RETURN
5260 REM **LUGAR 17**
5270 PRINT "Estas en el
      patio"
5280 LET N=1: LET E=1: LET
      S=0: LET O=1: RETURN
5290 REM **LUGAR 18**
5300 PRINT "Estas en el
      jardin"
5310 LET N=0: LET E=0: LET
      S=1: LET O=1: RETURN
5320 REM **LUGAR 22**
5330 PRINT "Estas en el
      armario"
5340 LET N=1: LET E=0: LET
      S=0: LET O=0: RETURN
5350 REM **LUGAR 24**
5360 PRINT "Estas en la sala
      del trono"
5370 LET N=1: LET E=0: LET
      S=0: LET O=0: RETURN
  
```

No te preocupes por el abundante uso de las sentencias REM y toda la memoria que consumen. En esta primera etapa del desarrollo del programa es muy importante saber qué es lo que hace cada parte del programa, o a qué número de lugar se refiere una descripción particular. Siempre puedes quitarlas más adelante.

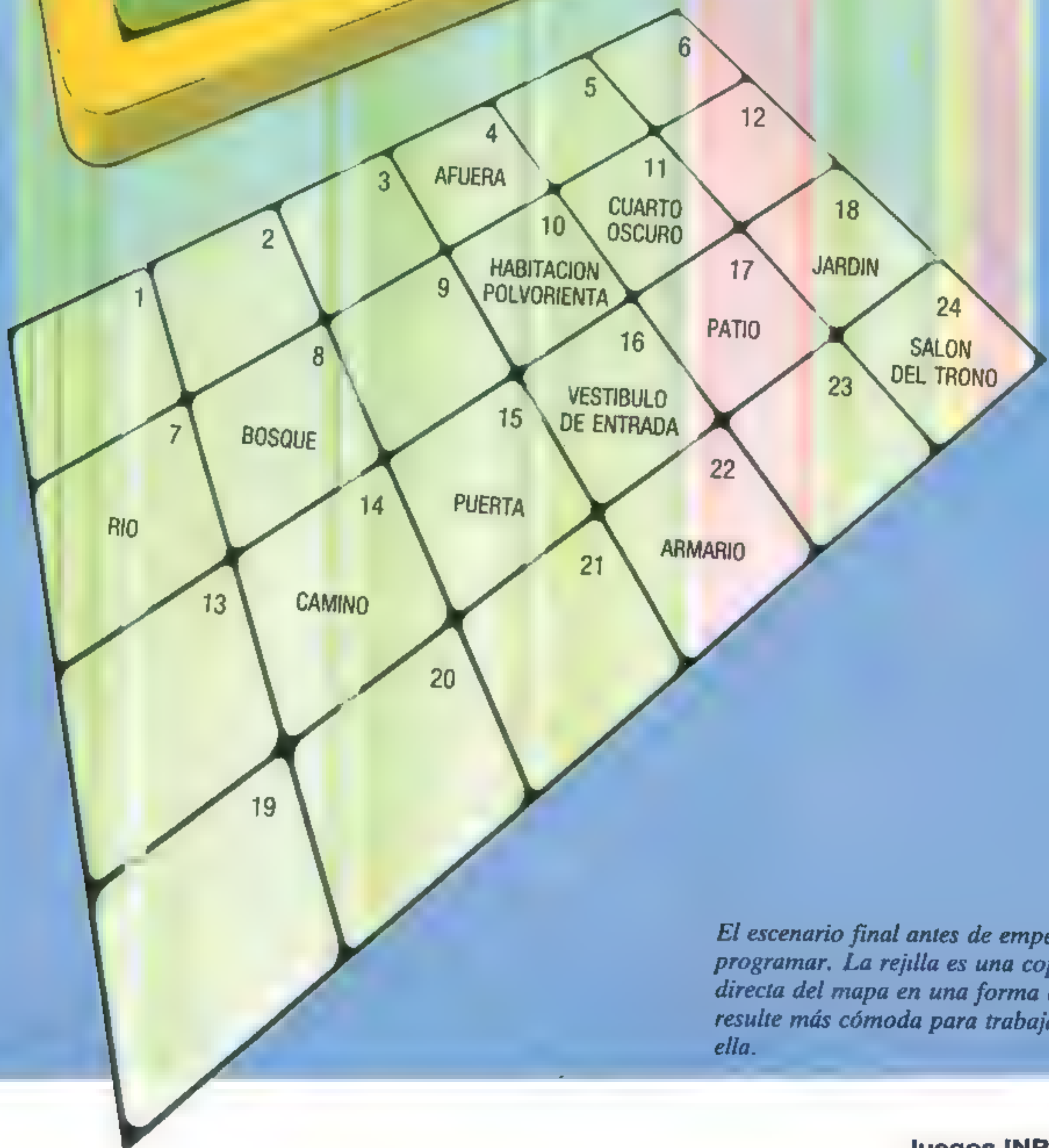
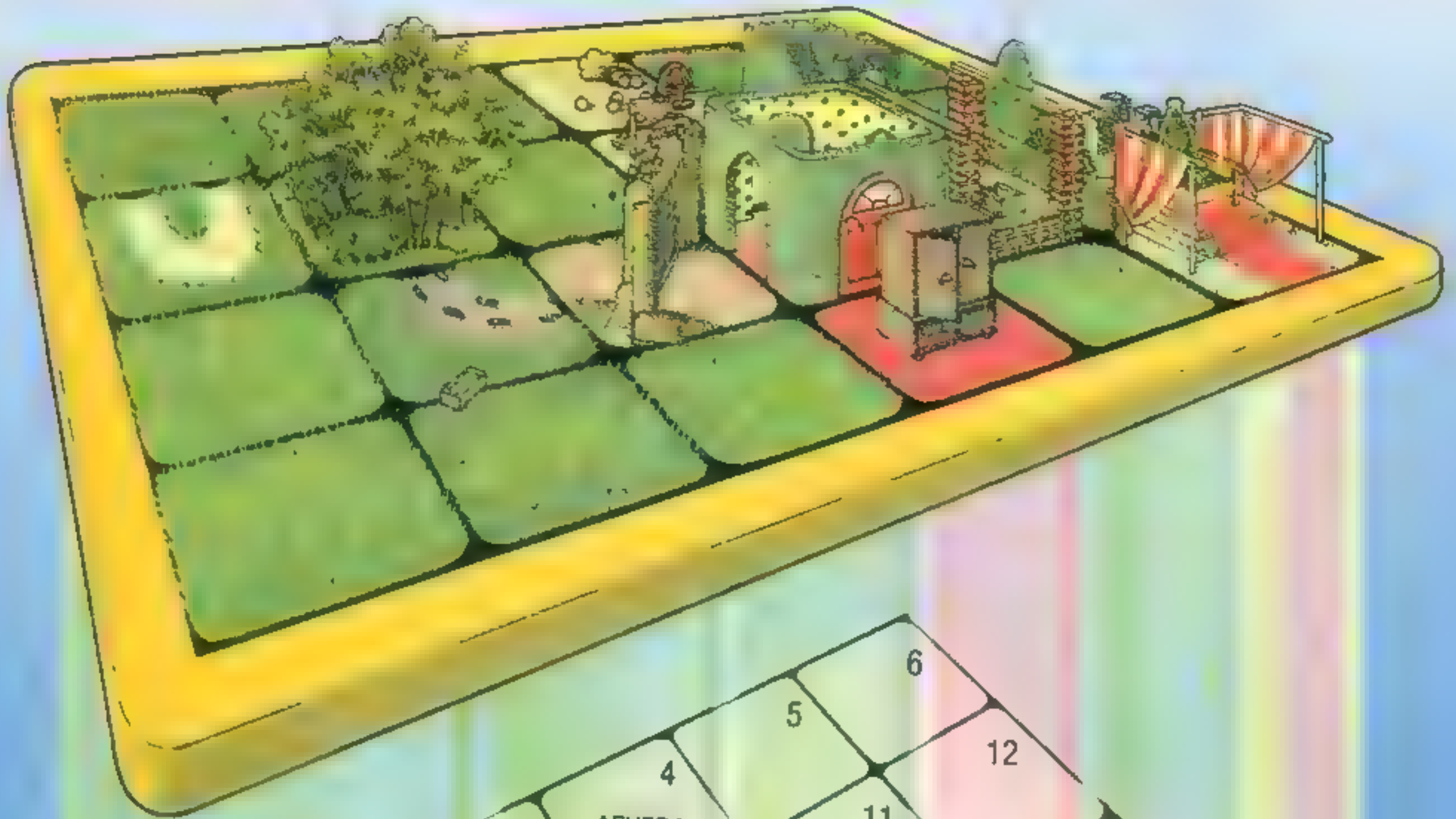
Después de cada línea con la descripción de un lugar, hay otra línea que contiene información sobre sus posibles salidas. Las variables N, S, E y O significan Norte, Sur, Este y Oeste. Estas variables pueden tomar uno de dos valores posibles: 0 significa que no hay salida en esa dirección, mientras que 1 significa que hay salida.

Finalmente, después de cada sección de este programa hay un RETURN, debido a que cada descripción de lugar se llama mediante una sentencia GOSUB.

La sentencia IF ... THEN que hay en la sección del cuarto oscuro es para comprobar si el aventurero tiene la lámpara, pero la descripción de las variables se comentará más adelante cuando nos ocupemos de los objetos.

En el próximo capítulo veremos la manera en que se mueve el aventurero por los distintos lugares.

PROGRAMACION DE JUEGOS



El escenario final antes de empezar a programar. La rejilla es una copia directa del mapa en una forma que resulte más cómoda para trabajar con ella.

UNA AVENTURA MOVIDA

Una parte de la diversión de los juegos de aventuras es la posibilidad de explorar mundos nuevos y extraños sin salir de casa. Veamos la manera de iniciar estas exploraciones.

Ahora que ya has tecleado un programa que contiene todas las descripciones de los ambientes, es el momento de hacer que el aventurero pueda explorarlos, desplazándose de un lugar a otro. Tienes que prever todos los

movimientos posibles en cada punto, y fijar unos límites, basándote en el grado de dificultad y en las pistas que se van recogiendo a medida que progresa la aventura.

En esta ocasión, para ampliar el desarrollo de tu programa, presentaremos rutinas con la descripción correcta de los ambientes, junto con las posibles salidas a los mismos. Al jugador se le pedirá que introduzca respuestas, y aprenderás la manera de escribir una

- PRESENTACION DE DIRECCIONES Y DESCRIPCIONES
- TRATAMIENTO DE LAS INSTRUCCIONES
- MOVIMIENTO A TRAVES DE LA AVENTURA

EL PUNTO DE PARTIDA

Lo primero que el ordenador tiene que saber es dónde está el aventurero en cada momento del juego. Para hacer esto el programa mira al valor que toma una variable llamada L (de lugar). A esta variable se le asigna un valor que corresponde al ambiente en que se encuentra el aventurero después de cada movimiento.

Para empezar pues la aventura, tienes que decirle al ordenador dónde quieres que se encuentre el aventurero en el momento de empezar.

Aquí tienes la primera parte de un programa que se encarga de esto. Carga (con LOAD) la sección de programa que tecleaste la última vez, y añádele las siguientes líneas:

```
100 CLS: LET DA=0: LET TA=0:
    LET LA=0
270 REM **posicion inicial**
280 LET L=15
290 GO TO 330
```

El número 15 corresponde al lugar en que se encuentra la puerta de la ciudad escondida. Si quieres que la aventura comience en otro lugar, basta con que modifiques el valor de L. En seguida veremos la forma de modificar el valor de L durante el juego, según el lugar atravesado. Pero antes de que el aventurero pueda empezar a moverse, hay que decirle al ordenador hacia dónde tiene que ir.

RESPUESTAS

Para que el ordenador pueda entender adecuadamente tus respuestas, y



PROGRAMACION DE JUEGOS

actuar en consecuencia, tienes que proporcionarle una lista de todas las palabras que puede reconocer.

En esta etapa del desarrollo, basta con que reconozca las cuatro direcciones NORTE, SUR, ESTE y OESTE. Podemos meterlas en una matriz R\$, que contiene los datos (DATA) de cada respuesta direccional.

```
110 REM **preparac. matrices  
    de respuestas**  
120 DIM R$(19,40): DIM R(19)  
130 FOR K=1 TO 4: READ R$(K)  
    ,R(K): NEXT K  
150 DATA "NORTE",1,"SUR",1,  
    "ESTE",1,"OESTE",1
```

En la línea 120 se dimensionan las matrices, de modo que contengan to-

das las respuestas necesarias para el juego. Dado que por el momento sólo necesitas utilizar las direcciones, sólo se emplearán los cuatro primeros elementos de las matrices R\$ y R. El bucle FOR ... NEXT de la línea 130, va desde uno hasta cuatro, leyendo tanto en R\$ como en R. Las direcciones y sus números están en las sentencias DATA de la línea 150.

Pero evidentemente esta información no le sirve para nada al jugador, a menos que el ordenador le diga previamente dónde se encuentra.

LA BUSQUEDA DE UN LUGAR

Para que el aventurero pueda saber adonde ha ido a parar después de cada

uno de sus movimientos, hay que suministrarle una descripción de los lugares. Estas descripciones ya las tienes tecleadas; sólo te hace falta una rutina para extraer la descripción que corresponda al valor de la variable L, el número de lugar. Aquí es donde resultan útiles las líneas REM que introdujiste anteriormente.

Pero, antes de escribir la rutina, tienes que añadir las siguientes líneas:

```
20 DIM G(11,4): POKE 23658,8  
30 FOR N=1 TO 4: FOR M=1 TO  
    11: READ G(M,N): NEXT M:  
    NEXT N  
40 DATA 0,0,0,5020,0,0,5050,  
    5080,0,5110,0  
50 DATA 5140,0,0,5180,5210,  
    5240,5270,5300,0,0,0
```




```

60 DATA 0,5330,0,5360,0,0,0,
  0,0,0,0
70 DATA 1010,1150,1240,1310,
  1410,1460,1500,1360,1080,
  1550,3110
300 REM **encuentra lugar**
310 CLS
330 IF L<11 THEN GO SUB G(L,
  1): GO TO 400
340 IF L<21 THEN GO SUB G(L-
  10,2): GO TO 400
350 IF L<26 THEN GO SUB G(L-
  20,3)
  
```

Antes de escribir este tipo de rutinas, tienes que asegurarte del número correspondiente a cada descripción de lugar. A partir del lugar número 1, tienes que ir haciendo una lista de los números de líneas de cada descripción. Si para alguno de los ambientes no hay ninguna descripción especial, asígnale un 0. En nuestra aventura, por ejemplo, no hay descripciones para los lugares 1, 2 y 3, pero sí la hay para el lugar 4.

Ahora que tienes la lista de los números de líneas, puedes empezar a escribir la rutina. Los números de las líneas se introducen en una matriz, como en el ejemplo que hemos visto antes.

Como habrás visto, se utiliza el valor de L para elegir un elemento de la matriz preparada por las líneas 20 y 30. En la matriz figuran unos cuantos números más que lugares. Todos los números extras son ceros que no influyen en el funcionamiento del programa, si bien son necesarios, ya que la matriz se usa para llamar a otras secciones del programa; en un próximo capítulo de nuestro coleccionable nos ocuparemos de ellos.

Finalmente, observa que el POKE

de la línea 20 sirve exclusivamente para que la máquina funcione con mayúsculas, a fin de que las respuestas que da el aventurero coincidan con las que están guardadas en R\$.

PRESENTACION DE LAS DIRECCIONES

Aparte de conocer las descripciones de los lugares, el jugador de la aventura querrá saber qué salidas tiene. Como en cada lugar no le será posible moverse en todas direcciones, el programa debe tener alguna forma de controlar las informaciones asociadas con la dirección, es decir las variables N, S, E y O. Las siguientes rutinas di-

rán al aventurero qué direcciones son factibles:

```
390 REM **direcciones**
400 IF DA<>1 THEN PRINT
    "puedes ir";
410 IF N>0 THEN PRINT TAB 11
    ;"NORTE"
420 IF E>0 THEN PRINT TAB 11
    ;"ESTE"
430 IF S>0 THEN PRINT TAB 11
    ;"SUR"
440 IF O>0 THEN PRINT TAB 11
    ;"OESTE"
```

La rutina se limita a comprobar el valor de las variables N, S, E y O, basándose en el mapa de direcciones. Si el valor de las variables es mayor que

cero, la dirección es posible y se presenta la correspondiente salida.

Esta rutina puede ser incorporada tal como está en cualquier aventura basada en una retícula de cuadrados.

INSTRUCCIONES

Ahora que el aventurero ya conoce las direcciones que puede tomar, hay que hacerle algunas sugerencias. La siguiente sección de programa le preguntará al jugador ¿QUE VAS A HACER AHORA?

```
450 REM **instrucciones**
460 INPUT INVERSE 1;"y ahora
    que "; LINE I$
470 GO SUB 3010
515 GO TO G(I,4)
```

Se trata de una rutina de entrada muy sencilla. A la respuesta suministrada le llamamos I\$. El ordenador debe comprobar que la respuesta es correcta y actuar en consecuencia. La

**ESTAS EN LA
ENTRADA**

línea 470 envía el control a la subrutina de la línea 3010, que es la encargada de procesar la respuesta dada por el jugador.

```
600 REM **rutina de
    verifica**
610 LET IN=0: IF LEN Y$>LEN
    X$ THEN RETURN
620 FOR K=1 TO (LEN X$-LEN
    Y$+1)
630 IF Y$=X$(K TO K+LEN Y$-1)
    THEN LET IN=K: GO TO 650
640 NEXT K
650 RETURN
3000 REM **verifica comando**
3010 LET N$="": LET X$=I$:
    LET Y$=" ": GO SUB 600:
    LET I=IN
3020 IF I=0 THEN LET V$=I$:
    GO TO 3050
3030 LET V$=I$(TO I-1)
3040 LET N$=I$(I+1 TO)
3050 LET I=0
3060 FOR K=1 TO 19
3070 IF V$=R$(K, TO LEN V$)
    THEN LET I=R(K): LET I$
    =V$(TO 1)
3080 NEXT K
3090 RETURN
```

Esta rutina comprueba si I\$ contiene dos palabras. De ser así, llama V\$ a la primera y N\$ a la segunda. Hemos puesto V\$ por verbo, como COGER, MATAR, o LLEVAR, y también sirve para las direcciones como NORTE, SUR, ESTE Y OESTE. N\$ significa nombre, y vale para designar los objetos que aparecen en la aventura. En las líneas 600 a 650 hay una pequeña subrutina muy útil que sustituye a la función INSTR de otros ordenadores; su misión es comprobar si existe un espacio de separación en I\$ entre V\$ y N\$. Si se encuentra un espacio, la línea 3020 designa a las dos partes de I\$ con las etiquetas N\$ y V\$. Si no se encuentra espacio, la propia línea 3020 llama V\$ a toda I\$.

El resto de la subrutina (líneas 3060 a 3080) compara las respuestas obtenidas con las que contiene la cadena R\$, que es la que contiene las direcciones de las respuestas. Más adelante veremos cómo puede ampliarse

P y R

¿Entenderá el programa respuestas tales como NORTE o IR NORTE, además de letras solas, tal como N?

La rutina de comprobación de instrucciones, situada en las líneas 3.000 a 3.010, está escrita de forma que recibe palabras de una sola letra, a las que maneja separadamente de las otras instrucciones más largas.

Las instrucciones de una sola letra se llaman I\$ y más tarde en el programa se comprueba si I\$ contiene las letras N, S, E o W, permitiendo al jugador que responda con la forma abreviada de la instrucción, y haciendo que el juego resulte más rápido de jugar.

Sin embargo, no hay nada que te impida hacer adiciones que permitan al aventurero utilizar respuestas del tipo NORTE o IR NORTE. La próxima vez veremos cómo maneja el programa los verbos y los nombres.

Lo que hay que hacer es añadir en la línea las palabras completas de las direcciones o añadir IR a la lista de verbos y escribir una rutina de procesamiento de los nuevos verbos.

para contener una serie adicional de verbos. Si en la línea 3070 se detecta una coincidencia, se pone en I el correspondiente valor de R. En fases posteriores del programa la máquina puede saber si ha habido una coincidencia examinando si I es mayor que cero. La última parte de la línea 3070, toma la primera letra de V\$ y la llama I\$, que servirá después para el movimiento del aventurero.

Esta subrutina puede utilizarse casi sin limitaciones en cualquier juego de aventuras. Probablemente la única modificación que le hagas, estará relacionada con la longitud del bucle FOR ... NEXT de la línea 3060.

MOVIMIENTO

Para que el aventurero pueda explorar todos los lugares, basta que añadas al programa una rutina que transforme la variable de lugar L, según el contenido de I\$. Aquí la tienes:

```
1000 REM **rutina de
    movimiento**
1010 IF I$="N" AND N>0 THEN
    LET L=L-6: GO TO 310
1020 IF I$="E" AND E>0 THEN
    LET L=L+1: GO TO 310
1030 IF I$="S" AND S>0 THEN
    LET L=L+6: GO TO 310
1040 IF I$="O" AND O>0 THEN
    LET L=L-1: GO TO 310
1050 REM **si direccion no
    es posible**
1060 PRINT "¡Lo siento no
    puedes ir por este
    camino!!": GO TO 330
```

Recuerda que la aventura estaba basada en una retícula de seis lugares. El movimiento implica un cambio del valor de L, que depende del tamaño de la retícula. Moverse hacia el Norte o hacia el Sur equivale a sumar o restar 6 al valor de L para desplazarse una línea hacia arriba o hacia abajo en la retícula. Análogamente, el movimiento hacia el Este o hacia el Oeste es equivalente a sumarle o restarle 1 al valor de L.

Las líneas 1010 a 1040 comprueban las direcciones en I\$ y ajustan L en consecuencia. Sólo es posible el movimiento si hay una salida que coincide con I\$. Las salidas fueron definidas en las líneas inmediatamente siguientes a las descripciones de los lugares.

Si no existe salida en la dirección en que el aventurero quiere salir, la línea 60 presenta el mensaje ¡LO SIENTO! ¡NO PUEDES SALIR POR AHI!

Para utilizar esta rutina con una aventura diferente, el único cambio que deberás hacer tendrá que ver con el tamaño de la retícula. En tal caso, habrá que modificar las líneas 1010 a 1030, con arreglo a la retícula.

Almacena ahora el programa (con SAVE) dejándolo listo para ser utilizado en el próximo capítulo.

DYNAMIC: ALGO MUY ESPECIAL PARA TU ORDENADOR

SPECTRUM



SPECTRUM



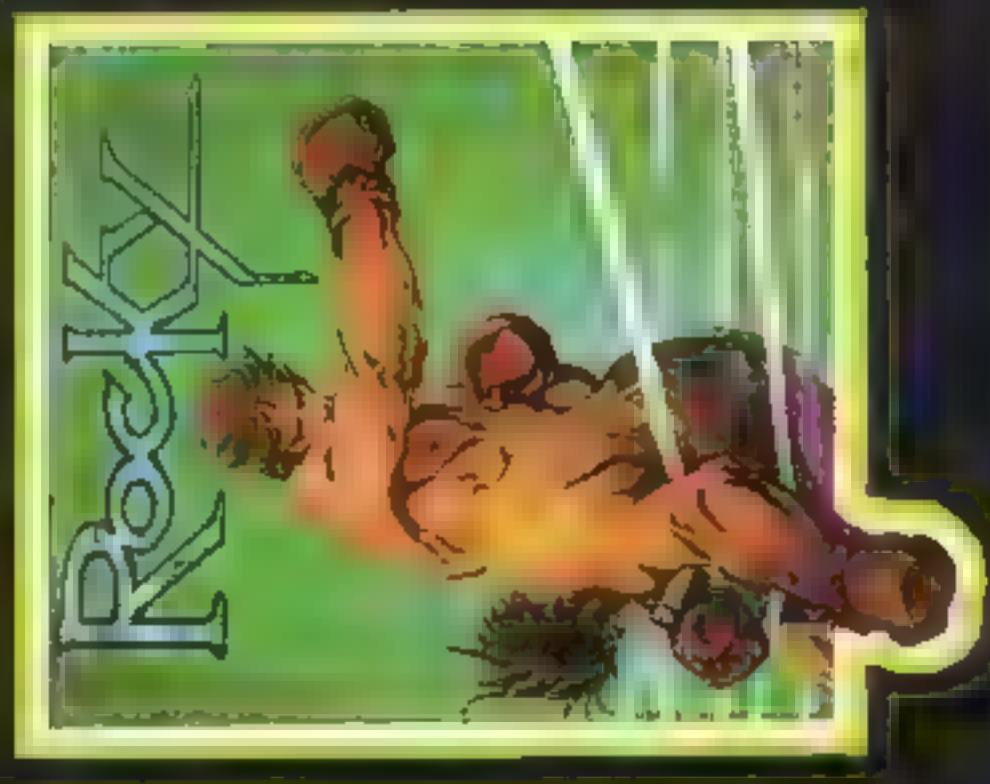
EDICION AGOSTO
A LA VENTA

SPECTRUM



NUEVO
A LA VENTA

SPECTRUM AMSTRAD



SPECTRUM AMSTRAD



SPECTRUM AMSTRAD



Tiendas y distribuidoras
(91) 447 34 10
Pedidos contra reembolso
(91) 715 00 67

DYNAMIC

EL CASTILLO ENCANTADO

Se trata de un juego cuyo objetivo es conseguir el tesoro que se encuentra en cada nivel, evitando ser capturado por el fantasma. Las puertas para pasar a otra habitación están ocultas, y para abrirlas hay que colocarse junto a la pared, enfrente de ellas.

Una vez que hemos obtenido el tesoro debemos volver a la habitación de la que partimos y encontrar allí la trampilla que está oculta en el suelo y que nos permitirá pasar al siguiente nivel. El juego se controla con las teclas de cursor.

El programa está escrito para el Spectrum 48K, aunque puede funcionar en el Spectrum 16K si la subrutina en código máquina se ubica en una dirección más baja. Para ello no es necesario modificarla, pero habrá que cambiar los POKES, adaptándolos a la nueva dirección.

Las letras y números subrayados deben ser tecleados en modo gráfico.

Borja Prieto Ursua

```

5 GO SUB 8500: GO TO 9000
1000 RANDOMIZE 0: GO SUB 9500
1199 REM BUCLE PRINCIPAL
1200 LET i$=INKEY$
1210 IF i$="" THEN GO TO 2500
1220 LET p=p+1: PRINT AT y,x;" "
1230 LET x=x-(i$="5" AND ATTR (y,x-1)<>2)+(i$="8" AND ATTR (y,x+1)<>2)
1240 LET y=y+(i$="6" AND ATTR (y+1,x)<>2)-(i$="7" AND ATTR (y-1,x)<>2)
1300 IF ATTR (y,x)=6 THEN GO TO 4000
1400 IF x=e AND (y=9 OR y=11) THEN LET zx=e: GO SUB 5000
1410 IF x=f AND (y=9 OR y=11) THEN LET zx=f: GO SUB 5000
1420 IF y=g AND (x=14 OR x=16) THEN LET zy=g: GO SUB 5500
1430 IF y=h AND (x=14 OR x=16) THEN LET zy=h: GO SUB 5500
1500 IF y=c AND x=d AND t=1 THEN GO TO 6000
2000 PRINT AT y,x; INK 5;"H": LET l=USR 50000
2500 IF a=10 OR b=15 THEN GO TO 7000
2510 IF in=1 THEN GO TO 7500
2520 PRINT AT a,b; INK 0; OVER 1;"F"
3000 GO TO 3000+(INT (RND*3)+1)*5
3005 LET a=a+(a<21 AND SGN (a-y)=-1)-(a>1 AND SGN (a-y)=1): GO TO 3020
3010 LET b=b+(b<30 AND SGN (b-x)=-1)-(b>1 AND SGN (b-x)=1)
3020 IF ATTR (a,b)=6 THEN LET in=1
3030 IF ATTR (a,b)=5 THEN GO TO 8000
3040 PRINT AT a,b; OVER 1; INK 7;"F"
3050 GO TO 1200
3999 REM CONTROLES JUEGO
4000 LET p=p+100: LET t=1: POKE 50011,100: LET l=USR 50000: POKE 50011,7: GO TO 1200
5000 PRINT AT 10,zx;" ": POKE 50001,10: LET l=USR 50000: POKE 50001,2
5005 RETURN
5500 PRINT AT zy,15;" ": POKE 50001,10: LET l=USR 50000: POKE 50001,2
5510 RETURN
6000 LET w=w-.05: LET p=p+r*25: LET t=0
6010 PRINT AT c,d; OVER 1; PAPER 4;"H": BEEP .1,0: BEEP .15,25: PRINT AT c,d; PAPER 4;" ": PAUSE 5
6020 GO TO 1000
7000 PRINT AT a,b; INK 2; OVER 1;"F": GO TO 3000
7500 PRINT AT a,b; INK 6; OVER 1;"F": LET in=0: GO TO 3000
7999 REM FINAL
8000 FOR a=1 TO 75: OUT 254,248: OUT 254,250: OUT 254,253: OUT 254,0: NEXT a
8010 FOR b=0 TO 3: FOR a=7 TO 0 STEP -1: PAPER a: CLS : NEXT a: NEXT b
8015 PRINT INK 7;AT 7,10;"HAS OBTENIDO:";AT 9,10,P;" PUNTOS"
8020 POKE 23658,8: INPUT "OTRA? (S/N)";a$
8030 IF a$="S" THEN GO TO 9000

```


[illegible]

VARIABLES: TODOS LOS MISTERIOS EXPLICADOS

■	¿COMO SE USAN LAS VARIABLES?
■	VARIABLES DE CONTROL
■	VARIABLES NUMERICAS Y CADENAS

Todas esas raras X e Y que aparecen en los programas son realmente muy sencillas de utilizar cuando se entiende lo que hacen. Ahora analizaremos la manera de utilizar las variables en tus propios programas.

Cuando quieres almacenar información en tu ordenador, tienes que decirle cómo puede identificarla. Si no lo haces así, el ordenador no podría encontrarla cuando vuelvas a requerirla.

Un tipo de información que se almacena temporalmente con mucha frecuencia son los números que incrementan o disminuyen su valor a medida que el programa avanza. Por ejemplo, podría tratarse de tu puntuación durante un juego o representar el número de espacios que un carácter se ha desplazado hacia la izquierda o la derecha en la pantalla o hacia arriba o abajo.

Muchas veces será simplemente un número cuyo valor cambia cada vez que el ordenador pasa por una sección del programa. Aquí tienes un sencillo ejemplo de esto:

```
10 LET X=0
20 LET X=X+1
30 PRINT X;" ";
40 FOR T=0 TO 10:NEXT T
50 GOTO 20
```

En la línea 50 se crea el bucle, haciendo que el programa se repita indefinidamente. Cada vez que llega a la línea 20, el número representado por X (que inicialmente había sido puesto a cero en la línea 10) se incrementa en una unidad.

La X de este programa es lo que se llama una **variable numérica**, ya que controla un número.

Una manera de representar la forma de trabajar de una variable numérica es imaginar una serie de cajas o

compartimentos, como los que se usan para clasificar la correspondencia. Cada caja tiene un nombre determinado. En un ordenador sencillo, los nombres serían letras del alfabeto. Cuando quieras almacenar un número, no tienes más que asignar ese valor a una de las cajas. Por ejemplo:

LET C = 25 (haz C igual a 25).

Esta instrucción almacena el valor 25 en la variable C. No todos los ordenadores requieren que se utilice la palabra LET (la llamada sentencia de asignación), pero hasta que seas más experto en programación, es una buena práctica incluirla. Con ello te resultará algo más fácil la lectura de programas o evitar llevarte chascos si cambias de ordenador.

Una vez almacenado el valor numérico, se puede usar en otras sentencias y cálculos. Por ejemplo, podrías poner:

PRINT C*4

o bien:

PRINT C/5

Estas sentencias darían las respuestas 100 y 5 respectivamente. Pero si te cleas:

PRINT C

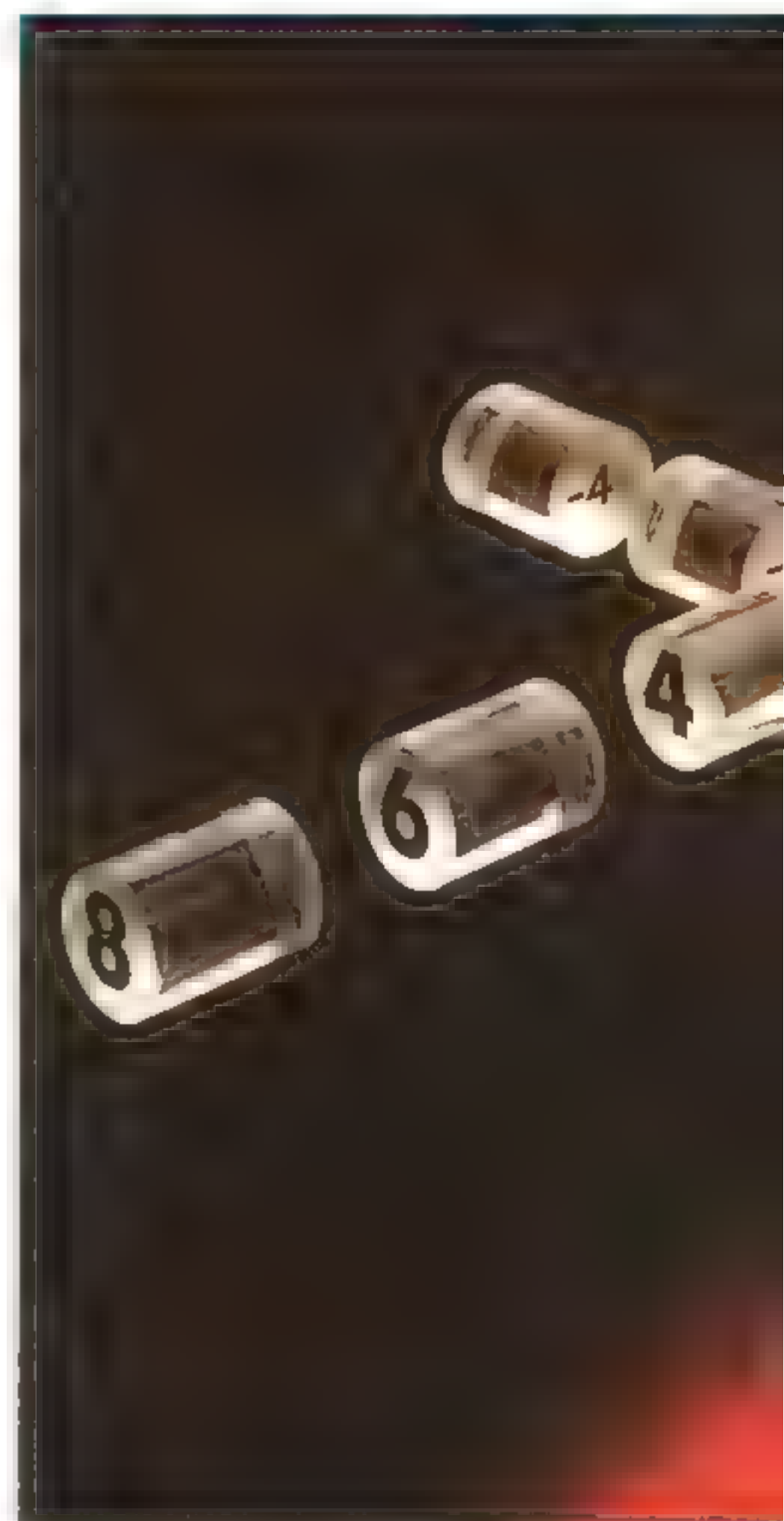
Verás que el valor de C ha permanecido sin alterar. Para cambiar el valor de C, por ejemplo, aumentándolo hasta 30, puedes hacerlo directamente tecleando

LET C=30

o tecleando

LET C=C+5

Naturalmente, esto es precisamente lo que hizo tu mini-programa original.



NOMBRES DE VARIABLES

Las combinaciones de caracteres permitidas como nombres de variables varían de un ordenador a otro. Más adelante veremos en un cuadro las principales reglas.

Pero una cosa que no te permitirá ningún ordenador es que el nombre de una variable comience con un número. Esto se debe a que sabe que

LET 7=14

no tiene sentido. Tampoco se te permite utilizar **palabras reservadas** del BASIC, como GOTO, THEN y AND.

VARIABLES EN ACCION

La mayoría de los programas que escribas utilizarán unas cuantas variables, con frecuencia interaccionando

unas con otras. Aquí tienes, por ejemplo, un corto programa que simula la acción de un surtidor de gasolina:

 para Spectrum

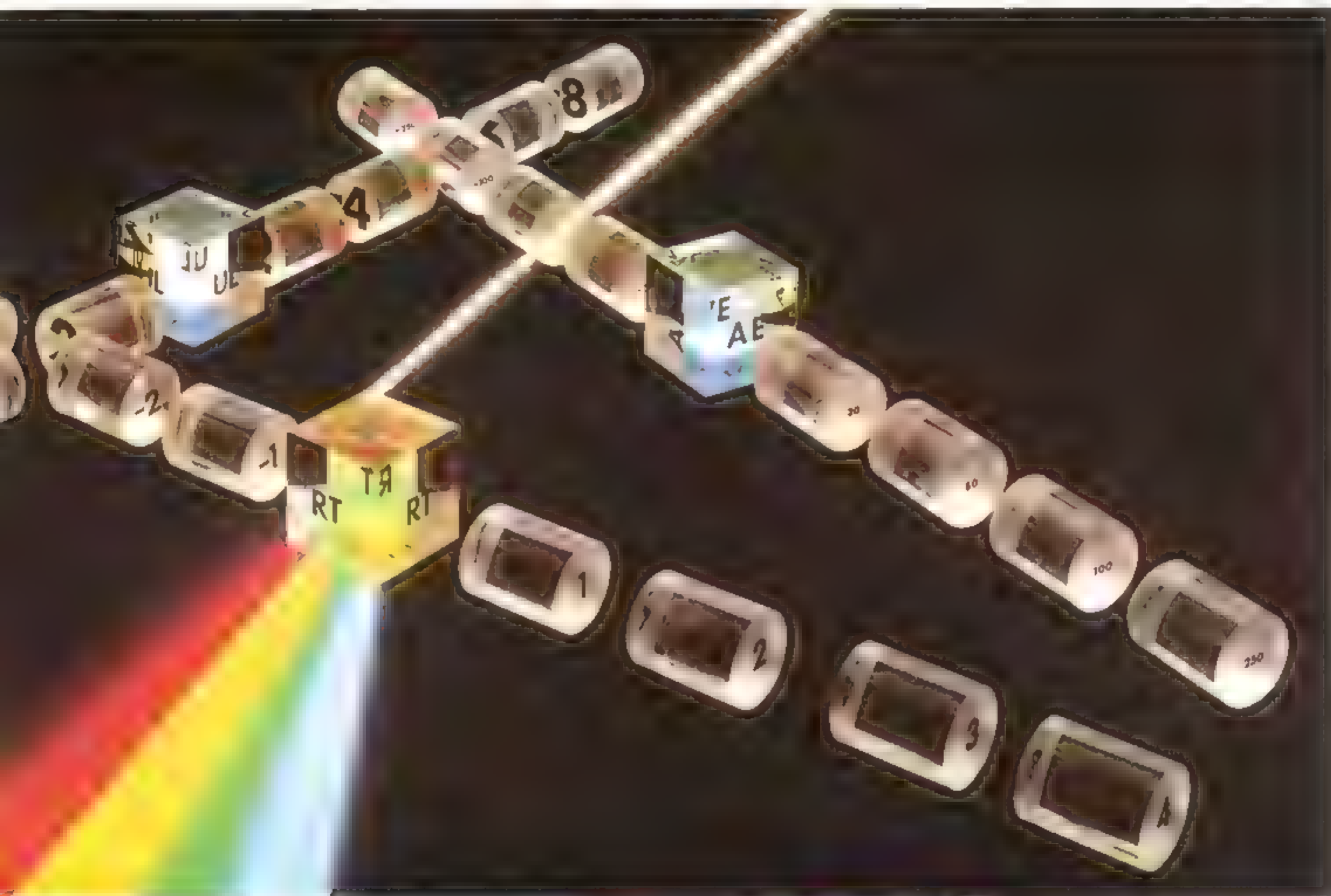
10 LET litros=0

20 LET pesetas=0

40 PRINT "TECLEA: (90)NORMAL
0 (97)SUPER"

50 INPUT a

60 IF a=90 THEN LET cant=
1/88



APRENDIENDO INFORMATICA

SPECTRUM EL MODUS OPERANDI

Si quieres que la informática deje de sonarte a chino, aquí tienes la solución. En tu propia casa, y a cualquier hora del día, puedes aprender informática con tu ordenador.

Investronica te ofrece ahora toda la información que necesitas, recogida en dos libros y cassettes, y apoyada por los mejores métodos de aprendizaje.

Ya tienes al alcance de la mano la más cómoda y fácil solución para adentrarte en el increíble mundo de los ordenadores.

Investronica te ofrece el modus operandi más completo para que aprendas informática.

5.000 pts.



SPECTRUM EL ORDENADOR CLASICO



investronica

Tomás Breón, 60. Tel. (91) 467 82 0 Télex 23399 IYOO E. 28045 Madrid
Camp. 80. Tel. (93) 21. 26 58-211 27 54 08022 Barcelona



```

70 IF a=97 THEN LET cant=
  1/91
80 IF a<>90 or a<>97 THEN
  GOTO 50
90 CLS
100 IF INKEY$="n" THEN LET
  pesetas=pesetas+50:LET
  litros=litros+cant
110 PRINT AT 10,12;"LITROS:
  ";litros;"[5*ESPACIO]"
130 PRINT AT 12,12;"PESETAS:
  ";pesetas;"[3*ESPACIO]"
150 IF INKEY$="f" THEN CLS:
  GOTO 10
200 GOTO 100
  
```

Conviene que experimentes con este programa hasta que entiendas completamente por qué tiene un conjunto tan completo de variables.

Lo que hacen algunas de ellas es evidente con sólo miraras. Por ejemplo, la variable A, sirve para ajustar el valor de una peseta al correspondiente valor de gasolina super, normal o extra.

A continuación, cuando pulsas N empieza a salir la gasolina por la manguera y el reloj empieza a contar. Pesetas, que había sido puesto a cero en la línea 20, comienza a incrementarse

de uno en uno (no se incluyen fracciones de peseta). Al mismo tiempo va aumentando la variable litros, que representa la cantidad de gasolina servida.

Cada vez que el programa llega a la línea 200, suponiendo que hayas mantenido pulsada la tecla N, se volverá a la línea 100 para mantener activo el reloj. Observa que no se vuelve a la línea 10, que pondría todo a cero otra vez.

Al ejecutar este programa observarás una cosa extraña: algunas de las fracciones decimales que resultan son muy largas. Esto no quiere decir que tu máquina tenga un defecto. Es simplemente una de las limitaciones de los ordenadores en cuanto a su capacidad de pasar de la aritmética binaria a decimal.

VARIABLES DE CONTROL

Igual que se utilizan variables en sentencias LET, también se emplean en los bucles FOR... NEXT, como ya has visto sin duda muchas veces. En este caso se llaman **variables de control**.

En algunos ordenadores, los carac-

teres que se pueden utilizar como variables de control son diferentes de los que se permiten para variables numéricas (ver cuadro más adelante). Por ejemplo en el **Spectrum** puedes poner:

```

LET puntuación = 0
LET mejor = 0
  
```

...Pero no puedes hacer más claro tu programa si utilizas un conjunto análogo de caracteres en lugar de una variable numérica, como en:

```
FOR tiempo = 99999
```

Debes utilizar una sola letra, por ejemplo T, en su lugar.

CADENAS DE CARACTERES Y VARIABLES DE CADENA

Una cadena de caracteres de BASIC es algo parecido a la cesta de la compra. Puedes llenarla con los más variados artículos, que manejas como una única unidad, sin tener que mover cada uno de ellos por separado.

Las cadenas de caracteres pueden contener casi de todo: letras, números, espacios, incluso caracteres gráficos. Siempre han de estar contenidas entre las comillas de ambos extremos. Aquí tienes unos cuantos ejemplos:

```
«NOMBRE Y DIRECCION DEL
CLIENTE»
«INTRODUZCA AHORA SU RES-
PUESTA»
  
```

```
«24 de enero»
```

```
«91-250 1000»
```

```
«28016»
```

El ordenador puede medir las cadenas de caracteres, es decir, contar el número de caracteres que la componen. También puede unir unas cadenas con otras (concatenar) o fraccionar una cadena en dos o más trozos cuando no se adapta a tu programa.

Pero el **contenido** de la cadena no le interesa para nada al propio ordenador. No puede sumaras, excepto en el

sentido ya indicado de unir una junto a otra, como tampoco puede multiplicarlas o dividir las.

Para probar esto —y también para desconcertar a tus amigos no poseedores de un ordenador— prueba este corto programa:

```
10 PRINT "2+2= ";
20 FOR N=1 TO 200:NEXT N
30 PRINT 1+2*2
```

Como puedes ver, el contenido de la cadena se imprime exactamente como lo has escrito. Sólo se hace un cálculo en la línea 30, imprimiéndose su resultado. (La línea 20 sirve para hacer que el ordenador «se rasque la cabeza» antes de dar una respuesta).

Si quieres usar la cadena solo una vez dentro de un programa, esto es todo lo que tienes que saber. Pero si quieres utilizar una cadena de caracteres más de una vez, puedes ahorrarte tiempo de mecanografiado y espacio de memoria, asignándole una etiqueta a tu cadena. Esta etiqueta se llama una **variable de cadena**. Su longitud varía de unos ordenadores a otros (ver cuadro), pero debe ir seguida siempre del signo \$.

Aquí tienes, por ejemplo, el esquema básico de un sistema de pedido automático:

```
10 LET A$="POR FAVOR,
ESCRIBE CUANTAS "
20 LET B$=" QUIERES"
30 PRINT A$;"HAMBURGUESAS";
B$;INPUT B
40 PRINT A$;"PATATAS FRITAS"
;B$;INPUT P
50 PRINT A$;"BEBIDAS";B$;
INPUT C
60 PRINT "GRACIAS, ME DEBES "
```



```
;B*180+P*85+C*110;
"PESETAS"
```

```
10 LET A$="POR FAVOR,
ESCRIBE CUANTAS "
20 LET B$=" QUIERES"
30 PRINT A$;"HAMBURGUESAS";
B$;
35 INPUT B
```



```
40 PRINT A$;"PATATAS FRITAS"
;B$;
45 INPUT P
50 PRINT A$;"BEBIDAS";B$;
55 INPUT C
60 PRINT "GRACIAS, ME DEBES "
;B*180+P*85+C*110;
"PESETAS"
```

No se trata de un sistema de pedidos especialmente bueno; sería mejor uno de tipo visual. Pero sirve para ilustrar cómo las cadenas pueden acortar un programa (aunque no tanto en el ZX81).

Por esta razón los sistemas de proceso de textos y de datos utilizan extensamente variables de cadena del tipo «número de cuenta del cliente», «precio por millar», «más el 15% de IVA» y otras cosas por el estilo que se presentan con mucha frecuencia.

Pero también puedes usar las cadenas en la programación de juegos. Supón que tienes una larga línea de sím-

LAPIZ OPTICO

SPECTRUM DIBUJA IPSOFACTO

Ahora que tienes tu magnífico Spectrum Plus, disfruta de él al máximo. Aumenta su capacidad y utilidad. ¡Aprovechal! Incorpórale el lápiz óptico. Con él puedes dibujar y borrar a mano alzada, mezclar textos, cambiar colores en dibujos, manejar tres pantallas de memoria auxiliares, almacenar dibujos en cassettes... Y además, puedes llegar a ampliar hasta 2, 4 u 8 veces la imagen en pantalla. Pruébalo y dibuja ipso-facto. Es una buena orden para tu ordenador.

8.125 pts.

SPECTRUM EL ORDENADOR CLASICO



investronica

Tomás Bretón, 60 Telf. (91) 467 82 10 Télex 23399 IYCO E 28045 Madrid
Camp. 60 Telf. (93, 21, 26 58 21) 27 54

bolos gráficos, por ejemplo, la pared de una mazmorra en un juego de aventuras. No tienes más que teclearlos una vez, «etiquetarlos» y ya puedes utilizarlos tantas veces como quieras.

Además puedes desplazar la cadena por la pantalla, como te muestra el siguiente programa:

```
10 LET b$="L39*ESPACIO]"
20 LET a$=b$+"L4*ESPACIO]"
  FELICES PASCUAS[L4*
  ESPACIO]" + b$
```

```
30 FOR n=1 TO 65
40 PRINT INK 2;AT 8,0;a$
  (n TO n+31)
50 PRINT INK 2;AT 12,0;a$
  (66-n TO 97-n)
60 BEEP .02,n/2
70 NEXT n
80 GOTO 30
```

Como muestra la línea 20, tienes una larga cadena: todos los espacios a partir de la línea 10, más el mensaje, más todos los espacios de nuevo, agrupados como un todo. En un próximo artículo veremos cómo se hace esto y cómo se «trocea» la cadena para que quepa en la pantalla. Pero entretanto

desearás cambiar el mensaje, asegurarte de que el nuevo tiene la misma longitud que el original. De no ser así, tendrías que modificar los números que intervienen en las líneas 40 a 55.

Un uso aún más común de las variables de cadena es el que se produce cuando el ordenador espera información que cambia constantemente. Por ejemplo:

```
10 PRINT "COMO TE LLAMAS"
20 INPUT N$
30 PRINT "ADIOS, ";N$
```

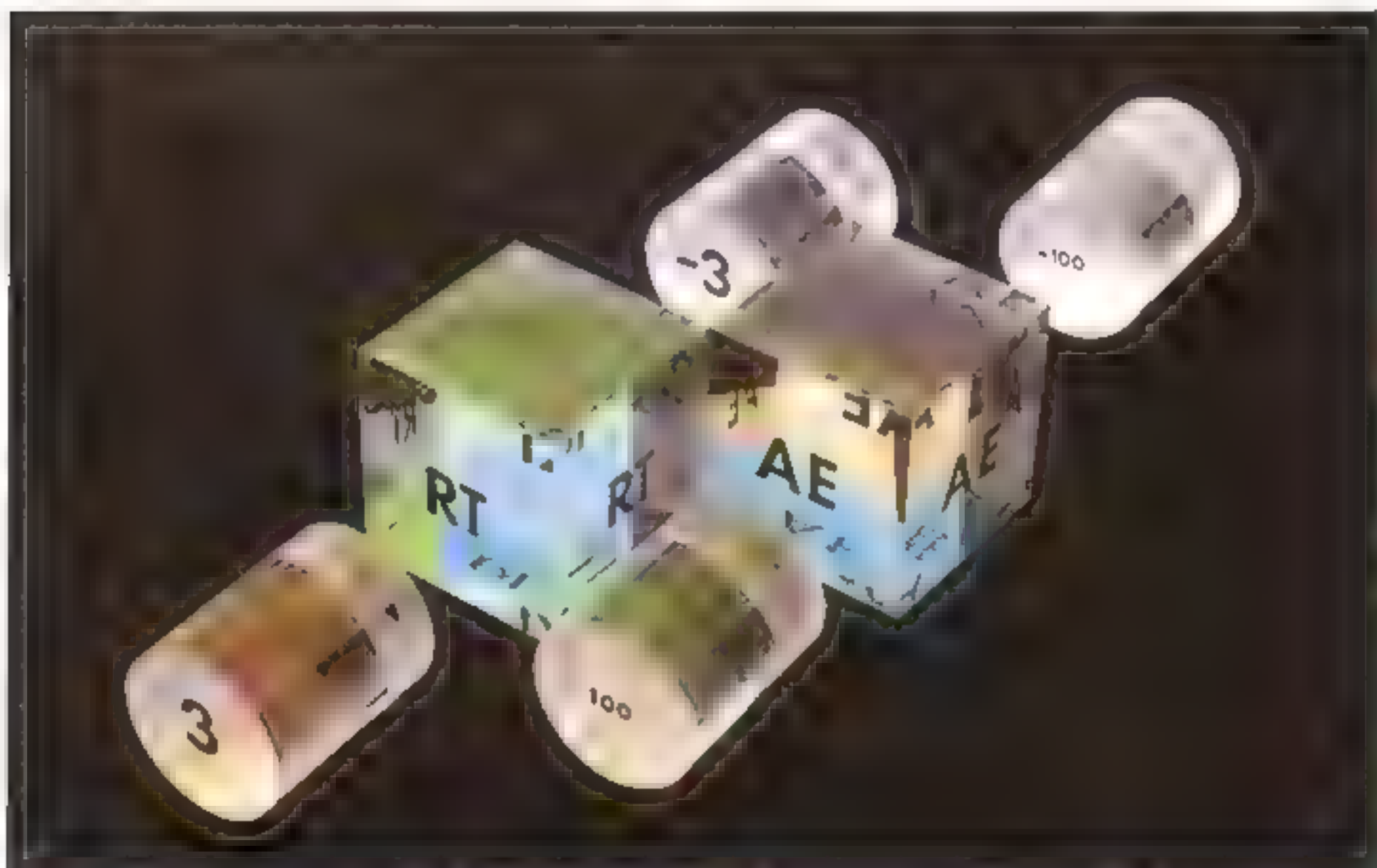
La línea 10 de este programa imprime la pantalla la cadena de caracteres ¿COMO TE LLAMAS? Seguidamente el ordenador espera a que introduzcas tu nombre, constituido por una serie de letras que forman otra cadena de caracteres, a la que hemos llamado N\$.

En la línea 30 se imprime la cadena «HOLA» y llama a la cadena N\$ que almacenó en la línea anterior. En la pantalla aparecerá.

HOLA, JUAN (o LUIS, o JAVI)

o cualquier otro nombre que hayas puesto.

Observa que a las cadenas ¿COMO TE LLAMAS? y HOLA no se les ha



GANADORES DE LOS MEJORES DE INPUT SINCLAIR

En el sorteo correspondiente al número 4 realizado entre quienes escribisteis mandando vuestros votos a LOS MEJORES DE INPUT han resultado ganadores:

Miguel Reyes Canales (Cádiz)
Abraham Salvador Luna (Córdoba)
Pedro García Fernández (Valladolid)
Miguel Angel Ruiz Coll (Madrid)
Oksana Rusler Kovalenko (Madrid)

Bibiana Fernández Ortega (Barcelona)
Fernando Sempere Belda (Valencia)
Pedro Morón Macías (Málaga)
Guillermo Torras Trenchs (Barcelona)
Alberto Goñi Sadalia (Guipúzcoa)

asignado un nombre como A\$ o COMO\$ u HOLA\$, ya que sólo aparecen una vez y el ordenador no necesita recordar lo que es HOLA. Cada vez que llega a la línea 20 se limita a leerla y a imprimir lo que contiene.

Sin embargo, el nombre introducido no necesariamente tiene que llevar una etiqueta. Puede ser diferente cada vez que se ejecuta el programa. También en este caso se almacena en la línea 20, pudiendo ser llamado por la línea 30.

Por último, aquí tienes una versión más corta de este programa, ligeramente «ensanchada» por razones de claridad. La forma corta es:

```
10 INPUT "COMO TE LLAMAS ";
   N$
20 PRINT "ADIOS, ";N$
```

CADENAS NULAS

También hay otro uso para las variables de cadena, que puede que hayas encontrado en la programación de juegos. Varía ligeramente de unas máquinas a otras, pero su apariencia es más o menos la siguiente:

```
20 IF A$="" THEN GO TO 10
```

Las comillas sin nada en medio constituyen la **cadena nula**. Significan «si la entrada es igual a nada»; en otras palabras. «si no se ha pulsado te-

VARIABLES: Lo que puedes hacer y lo que no.	
TIPO DE VARIABLE	SPECTRUM, ZX81
Variable numérica	Sin límite de longitud En el Spectrum se permiten mayúsculas y minúsculas, pero el ordenador no distinguirá entre ellas No permiten signos de puntuación Se permiten espacios
Variable de control bucles en FOR	Una sola letra p. ej.: A
Variable de cadena	Una sola letra seguida de \$, p. ej.: A\$

cla alguna», en cuyo caso vuelve a la línea 10 y espera hasta que se pulse alguna.

Esto evita que el ordenador pase por el programa tan rápidamente que simplemente el jugador no tenga ni siquiera tiempo de pulsar una de las teclas.

Una cosa muy distinta son las dos comillas con un espacio entre ellas. Si

modificas la línea de la forma siguiente:

```
20 IF A$="[ESPACIO]" THEN
   GOTO 10
```

El programa sólo volverá a la línea 10 si el jugador introduce un espacio, es decir si ha pulsado la tecla espaciadora.

INVESTICK E INTERFACE 2

SPECTRUM OPTIMUS LUDICUS

Si te gusta disfrutar a tope con tu ordenador, y sacarle el máximo partido, aquí tienes dos buenas ideas.

Incorpórale el Investick. El mando de hasta cuatro disparadores y ocho direcciones distintas de movimiento.

Y además, el Interface de Sinclair, con el que puedes conectar dos Investicks y utilizar los Cartuchos ZX de carga instantánea. Está el que te dará juego. Con Investick e Interface 2, tendrás juegos óptimos para ti y tu Spectrum, o para el nuevo Spectrum 128.



investronica
Tomas Breton, 60. Tel. (91) 467 82 10. Telex 23398 IYCO E. 28045 Madrid
Camp. 80. Tel. (93) 211 26 58-211 27 54. 08022 Barcelona



SPECTRUM EL ORDENADOR CLASICO

Super

ZS 48 052



SPECTRUM 48 K

El mejor de los programas de acción. Tú eres el Commando y tendrás que luchar contra un enemigo en superioridad. Necesitarás de todos tus reflejos y deberás mantenerte en constante tensión. ¡Empieza ya a disparar!

• Spectrum 48K • Commodore 64 • Amstrad •

Si deseas información y participar en los importantes sorteos que ZAFIRO celebrará durante el año... ¡ESCRIBENOS!



ZAFIRO SOFTWARE DIVISION
Paseo de la Castellana, 141. 28046 Madrid
Tel. 459 30 04. Tel. Barna. 209 33 65. Télex: 22690 ZAFIR E

Presenta
SIMULVISION

MAD
MAGAZINE'S OFFICIAL



ACCION Y ESTRATEGIA UNO O DOS JUGADORES
DOBLE PANTALLA DE ACCION SIMULTANEA



COMMODORE 64



Localiza las piezas del misil, montalo y escapa en el submarino antes de que el volcan entre en erupción. Otro espia intenta lo mismo que tu. ¿Podrás hacerlo?
1 o 2 jugadores. Dos pantallas simultáneas

• Spectrum 48K • Commodore 64 •

ventas

BOULDER DASH



ACCION Y ESTRATEGIA 16 CUEVAS DIFERENTES
324 PANTALLAS UNO O DOS JUGADORES

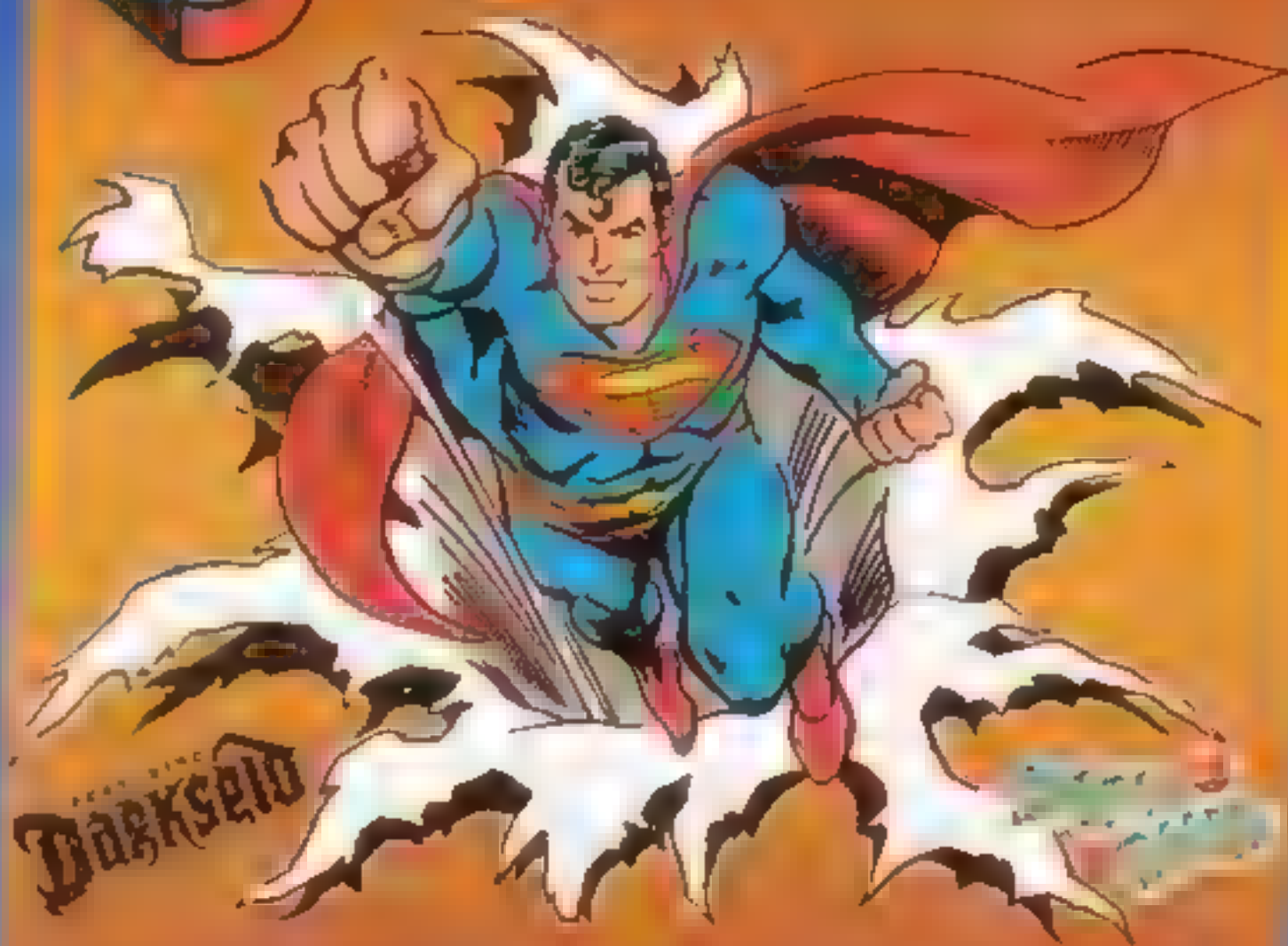


SPECTRUM 48K

Demuestra tu ingenio y rapidez. Un programa que crea fans
Vive el mundo subterráneo de Rockford en 5 niveles de
dificultad y 324 pantallas en 16 cuevas

• Spectrum 48K • Commodore 64 • Amstrad •

SUPERMAN EL JUEGO



• UNO O DOS JUGADORES • VARIOS JUEGOS EN UNO •
• CONSTANTE ACCION Y ESTRATEGIA •



COMMODORE 64

Más rápido que una bala, más potente que una locomotora... ¡Mira! ¡Arriba, en el cielo! ¿Es un pájaro?
¡No! Es... ¡Superman!
Vive las increíbles aventuras del Hombre de Acero
Ahora puedes volar y usar su visión calorífica y su
super-fuerza

• Spectrum 48K • Commodore 64 •

Si están agotados en tu tienda
habitual ¡¡LLAMANOS!!

Programas editados, fabricados y distribuidos en España
con la garantía Zafiro. Todos los derechos reservados.

INTELIGENCIA ARTIFICIAL Y SISTEMAS EXPERTOS

LAS SIGLAS I.A.
ESTADO ACTUAL DE
LA CUESTION
SISTEMAS EXPERTOS
QUE ES LA I.A.

La Inteligencia Artificial es uno de tantos temas de actualidad en el mundo de la informática de los que todo el mundo habla, pero de los que muy pocos saben algo que vaya más allá de los conocimientos más fundamentales.

Muchas veces nos hemos encontrado en alguna publicación con una serie de misteriosos conceptos que responden a las no menos misteriosas siglas de I.A. y que, «como por arte de magia», permiten a quien los maneja resolver todos los problemas sin que nosotros, sumidos en la ignorancia del principiante, podamos comprender de qué nos están hablando.

Es probable que ni tan siquiera sepamos a qué palabra corresponden dichas siglas, y si lo sabemos, es más probable aún que de ellas sólo sepamos eso: el nombre (**Inteligencia Artificial**).

Para muchos resulta inquietante la posibilidad de que los ordenadores dejen de comportarse como máquinas para pasar a adquirir cualidades humanas que, como en las novelas de ciencia-ficción, lleguen a superar la perfección de la inteligencia del hombre y a volverse contra su creador. Dejando a un lado la fantasía, y para apaciguar ánimos alarmistas, es conveniente remarcar las limitaciones actuales de la **cibernética**, y comprender un poco en qué consiste la **Inteligencia Artificial**, al menos para poder opinar con propiedad y enfocar el fenómeno desde el punto de vista correcto, es decir, en términos de «ayuda» y no de «competencia».

En esta serie de artículos trataremos de introducirnos en este apasionante mundo de la I.A., y de describirnos cuáles son las posibilidades de vuestro micro-ordenador en este campo, de forma que podáis elaborar vuestros propios programas o, al menos, saber algo más sobre un tema que

todo buen aficionado a la programación debe conocer.

Para ello, vamos a comenzar con una breve introducción histórica (como breve es la historia de una ciencia tan joven), y una panorámica de la situación actual de las investigaciones en España y el resto del mundo.

ESTADO DE LA CUESTION

Podemos considerar al matemático inglés **A. Turing** como padre de la **Inteligencia Artificial**, aunque esta disciplina tiene tantos «padres ilustres» que no tendríamos espacio para dedicar unas líneas a cada uno de ellos. Los primeros escarceos de la I.A. tal y como hoy la conocemos se remontan a mediados de los años cincuenta, cuando una serie de investigadores trataron de crear un modelo universal que llamaran «**General Problems Solving**», capaz de solucionar todos los problemas que se le plantearan. Este ambicioso proyecto, que aún en nuestros días resulta impensable, fracasó, provocando un cambio radical en los planteamientos.

A partir de este momento, se orientaron las investigaciones a la creación de sistemas capaces de trabajar óptimamente en materias específicas, eliminando así los insalvables obstáculos que planteaba el modelo anterior. Así nacieron los llamados «**Sistemas Expertos**», que en nuestro días han salido del campo excluyente de la investigación para pasar a ser objeto de aplicación práctica.

Un ejemplo de «**Sistema Experto**» muy cercano a nosotros lo tenemos en los programas de ajedrez y de «estrategia» que seguramente hemos manejado alguna vez. Más adelante trataremos de desvelaros cómo es posible que estos programas «razonen» por encima de las posibilidades del juga-

dor más experimentado, de forma que sólo los grandes maestros puedan derrotarlos.

En la actualidad, los principales problemas que obstaculizan el desarrollo de la I.A. son la limitación de las ramificaciones combinatorias que se alcanzan al tratar de analizar «todos los casos posibles» en la solución de un problema, y la imposibilidad de que el ordenador «comprenda» el lenguaje humano en todas sus diferentes interpretaciones, conociendo todas las connotaciones de una expresión y deduciendo cuál es la correcta según el contexto.

QUE ES LA INTELIGENCIA ARTIFICIAL

Para conocer, en una primera aproximación en qué consiste exactamente la **Inteligencia Artificial**, es necesario que primero sepamos cuáles son las características de un programa convencional. Aunque hayamos hecho muchos, es probable que hasta ese momento nunca nos hubiéramos planteado esta cuestión. En realidad, basta con ordenar un poquito nuestras ideas sobre cosas que ya sabemos:

En primer lugar, los programas convencionales se caracterizan por ser «rígidos» y «críticos», es decir, por poseer una estructura fija que hace siempre el mismo trabajo de forma repetitiva.

En segundo lugar, se caracterizan por su «dependencia decisoria», que se manifiesta especialmente en el hecho de que sólo responden ante casos previstos en el programa. En efecto, un programa convencional no puede alcanzar ningún tipo de conclusión si el programador no ha establecido previamente las premisas condicionales oportunas. El ordenador no decide por sí mismo, sino que actúa según unas pautas rígidas.

Otra característica muy importante es la forma del tratamiento de los datos. Estos funcionan en todo momento como algo externo al sistema, es decir, no pueden incorporarse al programa alterando su estructura.

Si alguna vez habéis programado, un juego por ejemplo, habréis comprobado lo tedioso que llega a resultar tener que explicar detalladamente al ordenador qué es lo que debe hacer exactamente en cada caso que se

le presente. Imagináos ahora lo que esto supone al diseñar un programa que «juegue al ajedrez», en el que los casos posibles son prácticamente infinitos, o un «wargame», donde las variables a tener en cuenta escapan a nuestra imaginación. Cuando los primeros genios de la cibernética se plan-

tearon este problema, llegaron a la conclusión lógica de que la única solución era tomar como modelo el funcionamiento de la mente humana. Si bien no es posible reproducir los secretos del raciocinio del hombre en una máquina, si pueden utilizarse sus fundamentos para crear un sistema «inteligente».

Dejando a un lado sentimientos y emociones, el hombre razona tomando como base no unas pautas previamente establecidas, como el ordenador, sino una serie de **conocimientos adquiridos**. De estos conocimientos, tomados de la experiencia, deduce reglas generales y crea conceptos nuevos basados en las relaciones lógicas que se establecen entre ellos.

Así, la **Inteligencia Artificial** incorpora a la informática el concepto de «aprendizaje», es decir, la utilización de la experiencia y la deducción lógica para auto-mejorarse y llegar a estadios de conocimiento cada vez más altos, tal y como el hombre hace.

Se trata, pues, de conseguir que la máquina «aprenda», incorporando los datos como elementos a partir de los cuales se formen una serie de relaciones complejas que vayan estructurando y desarrollando el programa.

Para comprender mejor esto, veamos un ejemplo de cómo un programa de I.A. imitaría el funcionamiento de la mente humana:

Para crear un programa «inteligente» que jugara al ajedrez, sería necesario que «pensara» las jugadas de la misma forma que su oponente humano, es decir, simulando los movimientos posibles y las posibles respuestas del adversario, remontándose en una cadena combinatoria hasta un límite determinado, valorando los resultados en cada caso, y eligiendo la jugada más favorable. La ventaja que aquí tendría la máquina sobre el hombre es su capacidad de memorizar y desarrollar cientos de jugadas imaginarias sin dejar escapar ninguna posibilidad.

Más adelante profundizaremos un poco más en estos temas. Por el momento, es importante que recordemos que la I.A. funciona en términos de «posibilidades». Si consultáis alguna bibliografía, tened en cuenta que al-



gunas traducciones no tienen en cuenta esta capital diferencia.

PROGRAMA

El programa listado al final de este artículo es la primera parte de otro mayor que concluiremos en próximos números. Su utilidad, a parte de ilustrar el tema que estamos tratando, es amplísima, ya que además de los usos que a continuación describimos, posee potencialmente otros muchos que os tocará a vosotros descubrir.

Esencialmente, es un programa con el que nos podremos comunicar a través de una serie de comandos en castellano, y que entre otras opciones, puede funcionar como hoja estadística, fichero, agenda, hoja de cálculo, etc.

En el próximo número describiremos más detalladamente sus posibilidades y la forma de comunicarnos con él. Os anticipamos que su propiedad más interesante es que no tenemos que atenernos a un «menú» numerado para indicar lo que queremos que haga, sino que podemos teclear directamente las órdenes (incluso varias a la vez). El programa «entenderá» lo que le pedimos y lo hará obedientemente.

Ahora puedes ir tecleando el listado, aunque no estará completo hasta la publicación de la última parte. A partir del próximo número, comenzaremos a desglosar las rutinas más importantes y su funcionamiento.

```

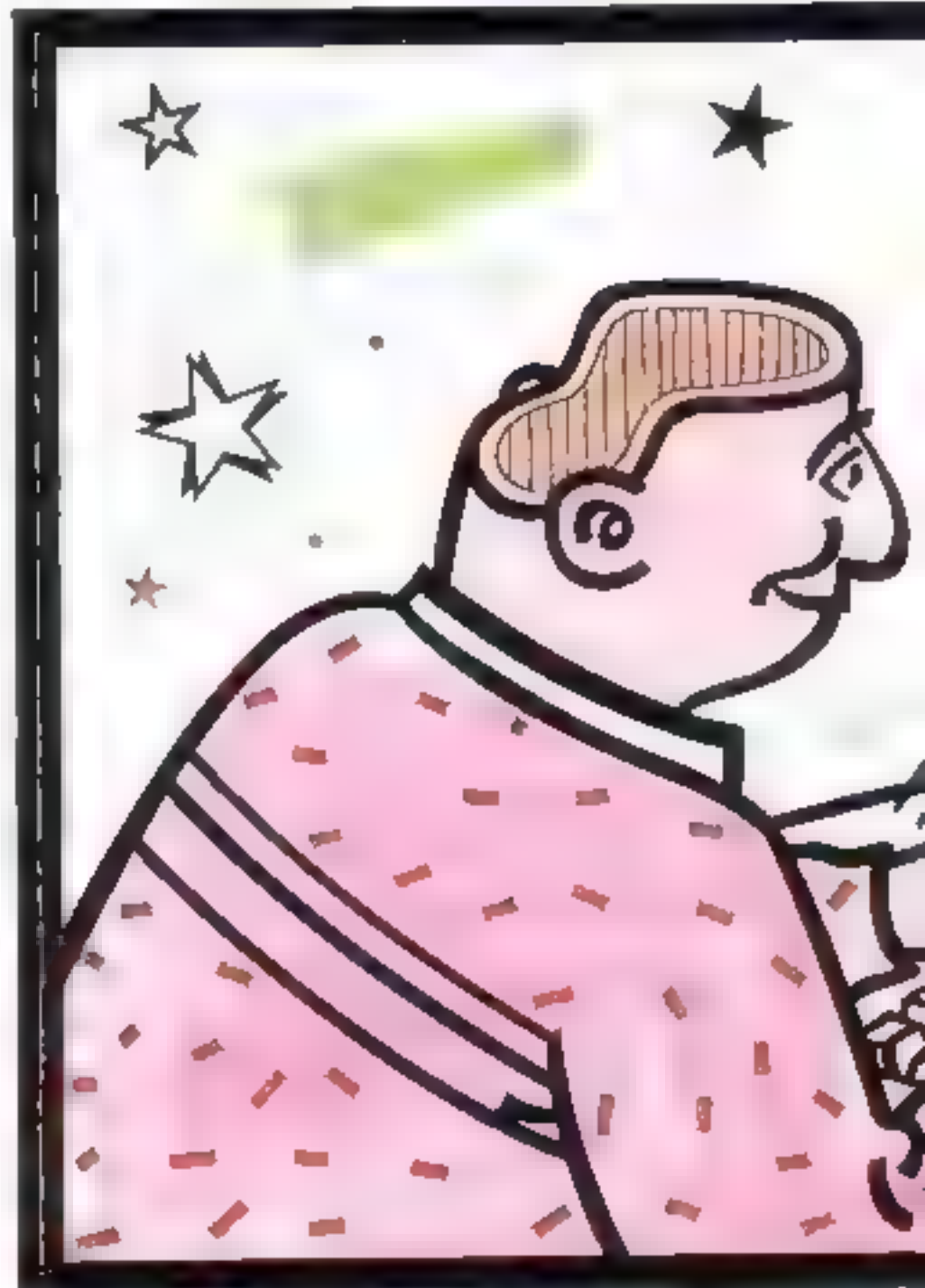
10 REM HOJA MULTIFUNCION
   ERNESTO DEL VALLE 85
   REVISTA 'INPUT'
20 REM INCIALIZADOR
25 GO SUB 9500
30 DIM a$(150,150)
40 POKE 23609,50
50 POKE 23658,8
60 INK 7: PAPER 0: BORDER 0:
   CLS
80 LET U$="HOJA": LET X=7:
   LET Y=13: GO SUB 90: LET
   U$="MULTIFUNCION": LET
   Y=9: LET X=9: GO SUB 90:
   GO TO 130
90 FOR N=1 TO LEN U$
100 FOR I=21 TO Y+N STEP -1
110 PRINT INVERSE 1;AT X,I;
   U$(N);INVERSE 0;+CHR$ 32
120 NEXT I: BEEP .01,20:
   NEXT N
125 RETURN
130 PRINT AT 11,9;"COPY E.
   DEL VALLE";AT 13,9;
   "REVISTA INPUT"
135 PRINT AT 21,2;"PULSA
   'ENTER' PARA COMENZAR."
140 FOR N=1 TO 1000
145 IF CODE INKEY$=13 THEN
   CLS: BEEP .1,30: GO TO
   170
150 BORDER 1: BORDER 2:
   BORDER 3: BORDER 4:
   BORDER 5: BORDER 6:
   BORDER 7: BORDER 0
160 NEXT N
165 CLS : BEEP .1,30
170 REM VARIABLES INICIO
180 LET L=0: LET M=1

```

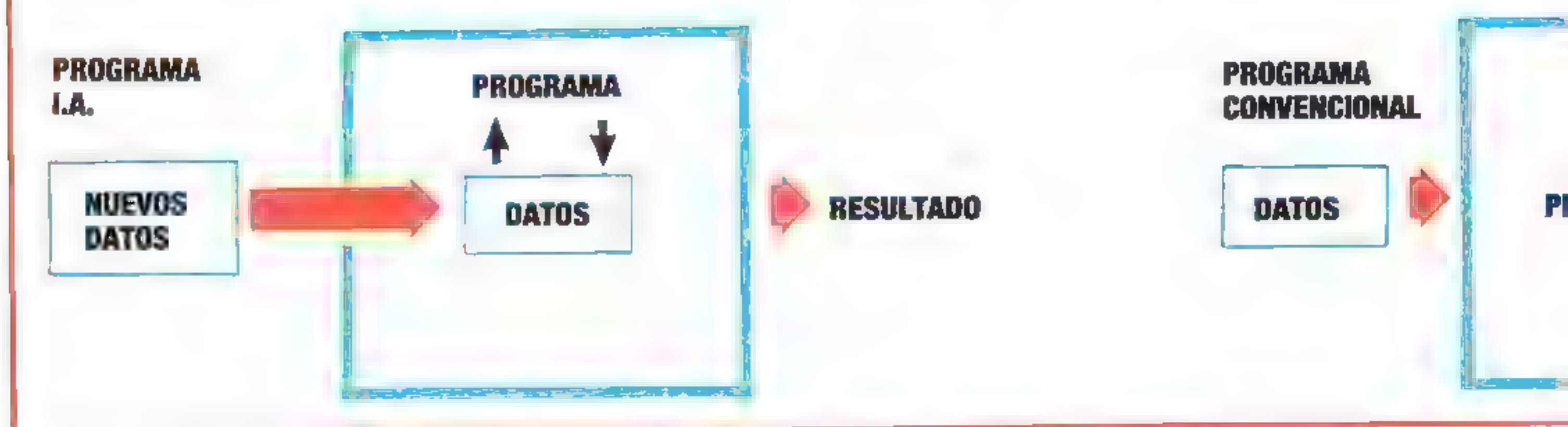
```

190 LET XX=0: DIM R$(110,32)
   : LET YY=0
195 GO TO 205
200 REM BIFURCADOR
202 BEEP .1,30
205 PRINT AT 21,0;"Esperando
   Ordenes"
210 INPUT LINE B$
220 GO SUB 500
250 IF N$="SUMA/" OR N$=
   "RESTA/" THEN GO TO 3500
260 IF N$="INTRODUCE/" THEN
   GO TO 1000
270 IF N$="INTRODUCE %/"
   THEN GO TO 1500
280 IF N$="MUESTRA/" THEN
   GO TO 2000

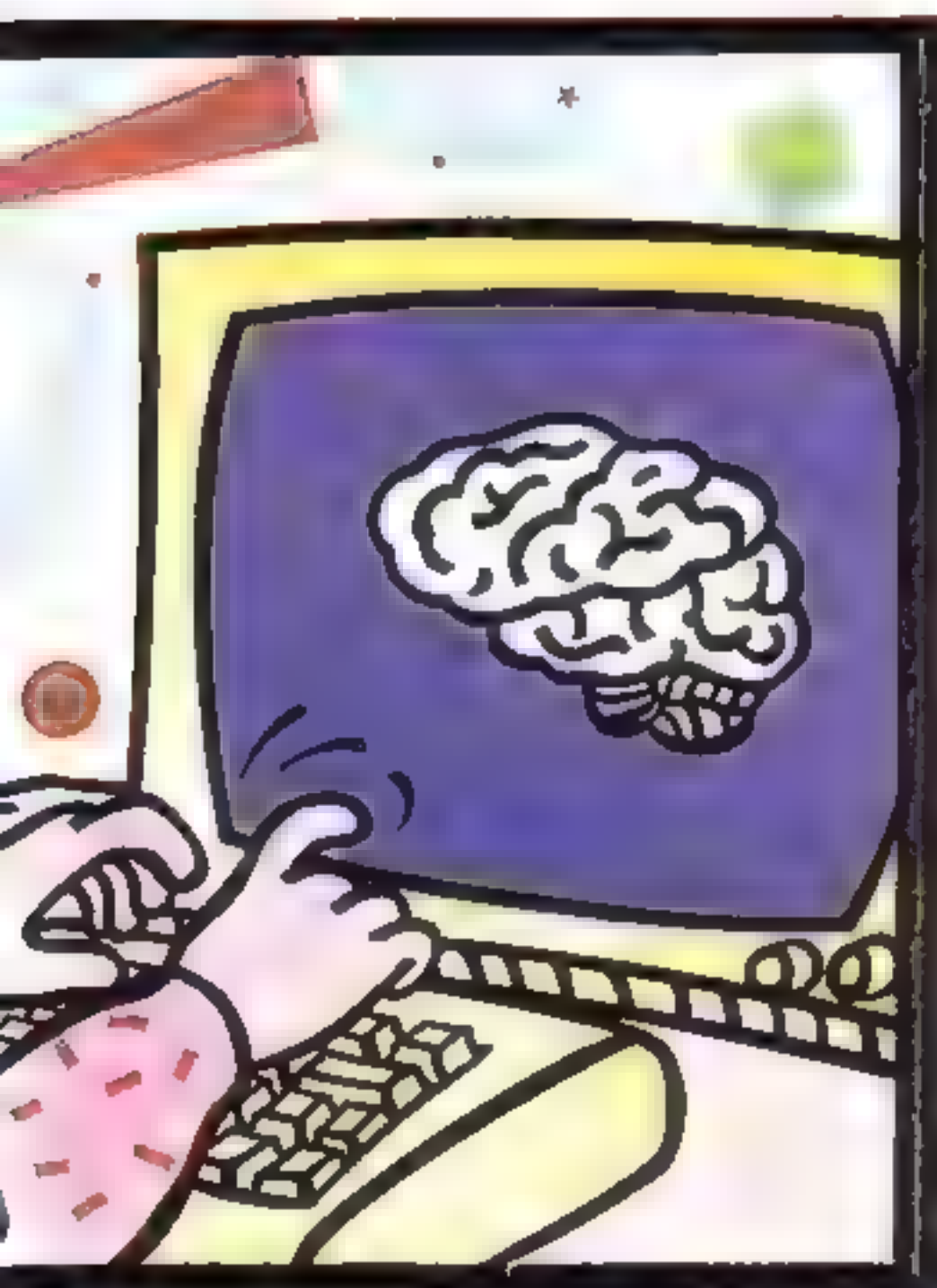
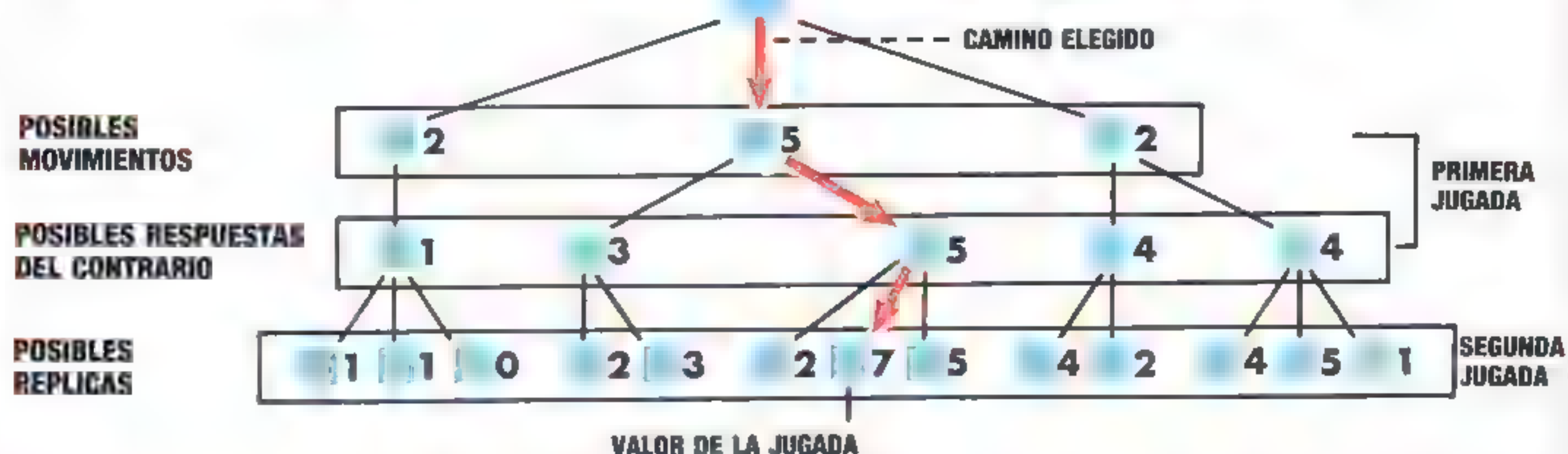
```



ESTRUCTURA COMPARADA DE UN PROGRAMA DE INTELIGENCIA ARTIFICIAL



DESARROLLO DE «ARBOL» EN UN PROGRAMA DE ESTRATEGIA



OTRO CONVENCIONAL

GRAMA



RESULTADO

```

290 IF N$="INTRODUCE #/" THEN GO TO 2500
300 IF N$="BORRA/" THEN GO TO 3000
310 IF N$="SALVA/" THEN GO TO 4000
320 IF N$="CARGA/" THEN GO TO 4500
330 IF N$="DESVIACION TIPICA /" THEN LET DT=1: GO TO 5000
335 IF N$="MEDIA/" THEN LET DT=0: GO TO 5000
340 IF N$="TEXTO/" THEN GO TO 8000
350 GO TO 200
500 REM ANALIZADOR
510 LET N$=""
520 FOR N=1 TO LEN B$
530 LET N$=N$+B$(N)
540 IF B$(N)=CHR$ 47 THEN RETURN
550 NEXT N
560 GO TO 200
1000 REM INTRODUCE DATOS
1005 LET YY=0: CLS
1010 LET N$="": GO SUB 550
1020 FOR F=1 TO L
1040 IF A$(F,1 TO LEN N$)=N$ THEN GO TO 1200
1050 NEXT F
1070 GO SUB 550
1080 LET L=L+1
1090 LET A$(L,1 TO LEN N$)=N$
1100 LET LL=L
1110 GO SUB 9000
1150 LET N$="": GO SUB 550
    
```

```

1160 IF B$(N)=CHR$ 47 THEN GO TO 230
1170 GO TO 200
1200 ETIQUETA YA COL.
1210 LET LL=F
1215 LET N$="": GO SUB 550
1220 FOR F=2 TO 150
1230 IF A$(LL,F)=CHR$ 32 AND A$(LL,F-1)=CHR$ 47 AND LEN N$+F<=150 THEN GO TO 1260
1240 NEXT F
1250 PRINT "NO HAY MAS ESPACIO BAJO LA[6* ESPACIO]ETIQUETA ";N$: PAUSE 0: GO TO 200
1260 LET F=F-1
1270 LET A$(LL,F TO F+LEN N$)=N$
1280 GO TO 1110
1290 GO TO 200
1500 REM INTRODUCIR CON %
1505 CLS : LET YY=0
1510 LET N$="": GO SUB 550
1520 LET L=L+1: LET A$(L,1 TO LEN N$)=N$
1521 LET V$=N$: LET OO=LEN N$: LET N$="": GO SUB 550: LET A$(L,OO+1 TO OO+1+LEN N$)=N$
1522 LET LL=L: GO SUB 9000
1525 GO SUB 1800
1528 LET O$="": LET P$="": LET P=0
1530 FOR F=1 TO LEN N$
2000 REM**ANTICIPO DEL PROGRAMA QUE IRA CON EL PROXIMO NUMERO
    
```


LOS MEJORES DE INPUT SINCLAIR

PUESTO	TITULO	PORCENTAJE
1.º	The way of exploding fist ..	23,8 %
2.º	Profanation	18,6 %
3.º	The Dambusters	11,9 %
4.º	Comando	8,2 %
5.º	Saboteur	8,2 %
6.º	Basketball International	7,4 %
7.º	West Bank	7,4 %
8.º	Panorama para matar	5,9 %
9.º	Super Test	4,9 %
10.º	Back to skool	3,7 %
		100 %

Para la confección de esta relación únicamente se han tenido en cuenta las votaciones enviadas por nuestros lectores de acuerdo con la sección «Los Mejores de Input».

Febrero de 1986



EJECUTIVOS DE ALTO NIVEL

Se busca presidente para dirigir empresa multinacional líder en el sector informático. Se requiere capacidad para tomar importantes decisiones. Se valorarán conocimientos contables y financieros. Imprescindible español a nivel de traducción. No es necesaria experiencia.

Si te atrae el mundo de los negocios, este programa de **ERE Informatique** puede servirte como plataforma de lanzamiento, ya que en tus manos está el control económico, financiero y social de una gran empresa. Como director de la misma estás facultado para variar el precio de los productos, su fabricación, gastos, inversiones, préstamos...

La empresa **Info-Gesti** está dedicada a la fabricación y venta de diversos productos informáticos: un microordenador, una unidad de *diskettes*, una impresora y dos programas, uno de contabilidad y otro de control de existencias. Además de estos productos también tiene en desarrollo un micro de la gama baja y un programa de gestión de ficheros que podrán ser lanzados como nuevos productos cuando lo estimes oportuno.

El número de empleados asciende a 115 repartidos en cinco talleres de producción, cinco servicios de ventas, un departamento financiero y servicios generales.

Trimestralmente, el Comité Directivo se reúne y, tras examinar los resultados del ejercicio, acuerda las nuevas decisiones en cada una de las áreas que integran la sociedad. Una vez fijadas todas las actividades, el programa simula las consecuencias económicas que producen dichas decisiones y presenta la nueva situación de la empresa.

En todo momento es posible solicitar información de cualquier área de actividad que puede ser relativa a los productos, balance, cuenta de pérdidas y ganancias, cuenta de explotación, ventas, avisos, etc.

Todas estas variables son susceptibles de ser alteradas por lo que es posible aumentar o disminuir el precio de los productos, variar los gastos en publicidad, provocar una mayor o menor producción e incluso contratar o despedir empleados.

Si quieres ampliar el negocio y la empresa goza de buena solvencia, no es difícil conseguir un préstamo que permita realizar inversiones para el futuro. También es necesario controlar las existencias y tener en cuenta las amortizaciones.



En el organigrama de la empresa el Presidente está en el vértice de la pirámide; de él dependen las tres direcciones generales: Producción, Marketing y Financiación.

El Director de Producción tiene tres actividades fundamentales: decidir las unidades a fabricar, aumentar o disminuir los empleados de la cadena de producción e invertir en nuevos equipos para mejorar el rendimiento. El departamento de Marketing fija los precios de venta, selecciona el número de vendedores, decide los

gastos en publicidad, el lanzamiento de un nuevo producto y la paralización del proceso de fabricación de un determinado producto.

Por último el Director Financiero debe conseguir los préstamos a largo y corto plazo, decidir los aumentos de capital, el importe de la amortización, la revisión salarial e incluso la contratación o los traslados de personal.

Si bien contamos con una gran mayoría de parámetros que en cierta forma son controlables, existen otros factores ajenos al sistema y que pueden provocar importantes alteraciones. Entre estas variables externas se pueden mencionar: la aparición de competencia, huelga laboral, dificultad para conseguir crédito, variación de los impuestos o tipos de descuento, alteración del mercado de divisas e incidentes en el proceso de fabricación. Cada una de ellas podrán mejorar o empeorar la situación de la compañía.

El programa permite, una vez finalizado un trimestre, almacenar en *cassette* la situación de la sociedad y poder continuar posteriormente el desarrollo del juego en el periodo en el que se quedó. Aquellos que disponen de impresora podrán listar cuanta información precisen y trabajar con una mayor independencia.

Es prácticamente imposible resumir en pocas líneas las enormes posibilidades, la gran variedad de factores que incluye el programa y las muchas horas que son necesarias para conseguir mantener una empresa medianamente saneada. En cualquier caso, la gran ventaja que tiene es que a pesar de tomar decisiones equivocadas, que pueden desembocar en suspensión de pagos, e incluso en la quiebra de la empresa, esto no repercutirá en tu *curriculum* e inmediatamente serás nombrado de nuevo Presidente. No todos tienen esa suerte.

¿ERES EL BUENO O EL MALO?



Tenemos ante nosotros un juego de guerra, aunque no tiene nada que ver con la estrategia, los

movimientos de tropas, etc.

Únicamente se trata de un cruel y encarnizado combate.

Repasemos la trama. El Dictador ha capturado como rehenes a las tropas enemigas. Lograr su rescate es el desafío.

La pantalla de presentación nos permite elegir uno de entre tres niveles de dificultad. Igualmente puede optar por uno o dos jugadores; es decir, que puedes perfectamente ser tú el Dictador que intenta la derrota de las tropas aliadas, encabezadas por tu padre o hermano.

Para intentar el rescate de las tropas recluidas se lanzarán las fuerzas paracaidistas desde el helicóptero, durante esa primera fase del juego.

Un enorme y destructivo cañón amenaza a los paracaidistas, escondidos tras un muro después de su toma de tierra. Una brecha en el primer muro te permite llegar a otro más próximo al cañón, primer objetivo que debe ser alcanzado para entrar a la fortaleza. El peligro de salir a campo abierto es el consabido de estar a merced de los proyectiles de la artillería del Dictador, los cuales habrá que burlar. Los supervivientes, si los hay, habrán de tomar finalmente el cañón y neutralizarlo. Por tanto la misión de estas fuerzas de choque es fundamental. La estrategia empleada debe consistir en que los hombres

salen por parejas, el primero procura atraer el fuego sobre sí, mientras el otro lanza una granada y sale corriendo hacia un lateral para esquivar los disparos. Cubierto el primer objetivo, llegamos a la segunda pantalla en la que el resto de la tropa debe atravesar el patio de la fortaleza. Con el cañón recién tomado al enemigo, cubrirás el ataque de tus hombres. Es imprescindible aniquilar a los cuatro adversarios que oponen toda su resistencia contra el avance. El más peligroso es uno que se esconde tras una trampilla móvil por todo su recorrido. Se desplaza con enorme rapidez y continuamente sale un enemigo que dispone bombas a dos pasos de tus hombres. Por ser el más conflictivo es el primer adversario que debes eliminar. Siguiéndole en peligrosidad está un automóvil que te persigue implacablemente y una vez alcanzado te desintegra sin piedad. Los dos últimos obstáculos son un tanque, que intenta aplastarte mediante el atropello, y un soldado escondido tras las murallas que dan al patio, por donde lanza mortíferas bombas.

Si logras llegar a la tercera fase, debes elegir una ruta de escape. Pilotas un helicóptero, te persiguen misiles, te atacan desde *bunkers*, desde trampas escondidas en el suelo lanzan pelotas. Un sinfín de peligros te obstaculizarán la entrada a la madriguera del Dictador. La cuarta etapa es un enfrentamiento personal entre los dos líderes: el comandante **Striker** y el Dictador. En una gruta les separa un precipicio por el que avanza el curso de un río. Entonces hablan los «poontas», que se lanzan mutuamente ambos contendientes con la intención de dar muerte al otro. El vencido cae y es arrastrado por las frías aguas del río. Aunque el juego no aporta nada nuevo del otro mundo en cuanto a los gráficos, la emoción está asegurada.



EL CASTILLO DE AVARS

Vas a penetrar en el mágico y maravilloso mundo de **Fairlight**. En esas lejanas tierras, un castillo encantado encierra grandes secretos. Allí hace muchos años el sol brillaba alegremente, pero unos extraños acontecimientos lo sumieron todo en la oscuridad. Sólo el castillo de **Avars** no sucumbió en la tragedia. Para descubrir su misterio debes introducirte en él y resolver el enigma de **Fairlight**.

El programa nos muestra el estado actual de la tecnología en gráficos tridimensionales y es el primero de una serie que la firma **The Edge** irá publicando sucesivamente.

Los objetos que van apareciendo tienen características reales, como peso, inercia e incluso un cierto grado de inteligencia. Entre ellos podemos mencionar: llaves, barriles, garrafas, pergaminos, pan, plantas,

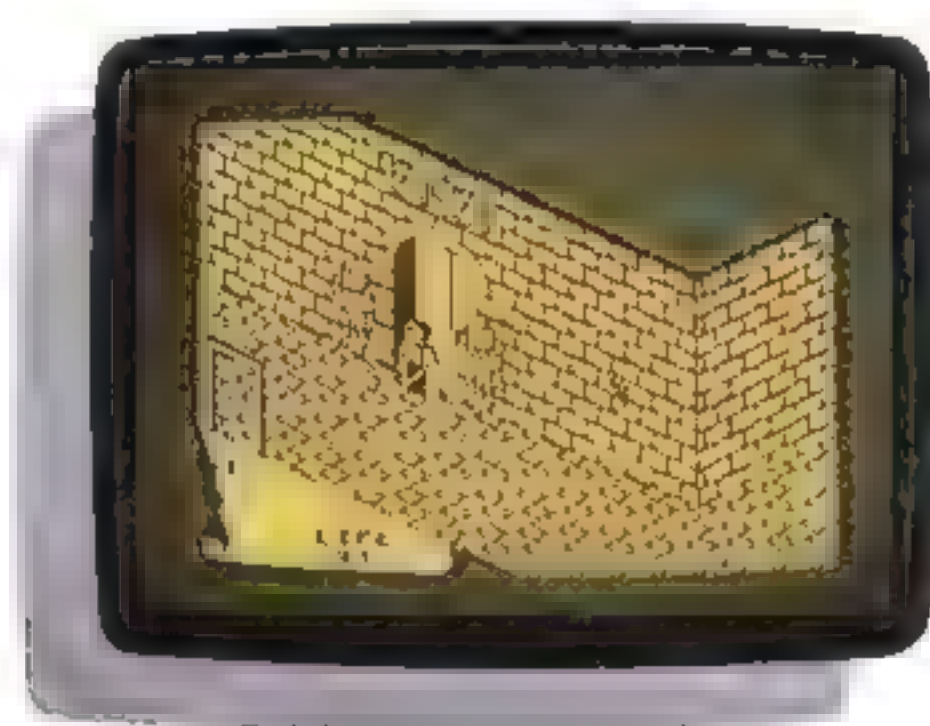
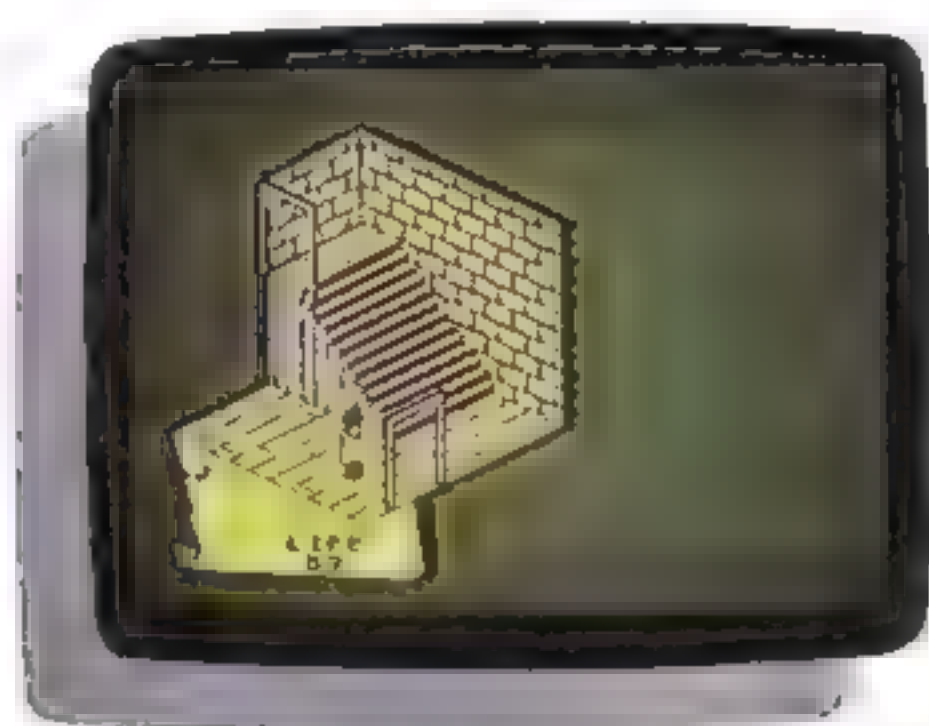


la que la dejamos. Eso quiere decir que si un enemigo te ha perseguido hasta la puerta o un objeto ha quedado en un determinado lugar, aunque salgas y entres todo continúa en la misma posición.

Las claves para resolver el problema están esparcidas por todo el castillo, así como en los textos que

con la ingestión de alimentos como pan y pollo, o bebiendo el contenido de la garrafa.

Te advertimos que te encontrarás a unos centinelas con cuyo contacto disminuirán tus defensas; evita enfrentarte con ellos, y si lo haces, procura eliminarlos cuanto antes. Además, merodea por las



balsas de oro... La mayoría tienen un uso específico, pero otros es mejor no encontrárselos.

Dispones de cinco bolsillos en los que guardar estos objetos que surgirán, pero piensa que si tomas uno demasiado pesado, no podrás transportar ningún otro.

En la mayoría de los juegos, cuando en una pantalla tenemos dificultades, salimos de ella sabedores que si volvemos a entrar los problemas serán menores, ya que los enemigos siempre retornan a su origen. Pues bien, en este caso el truco no te servirá, ya que cada habitación permanece en la misma situación en

acompañan al *cassette* y en los que encontrarás pistas para desarrollar tu misión.

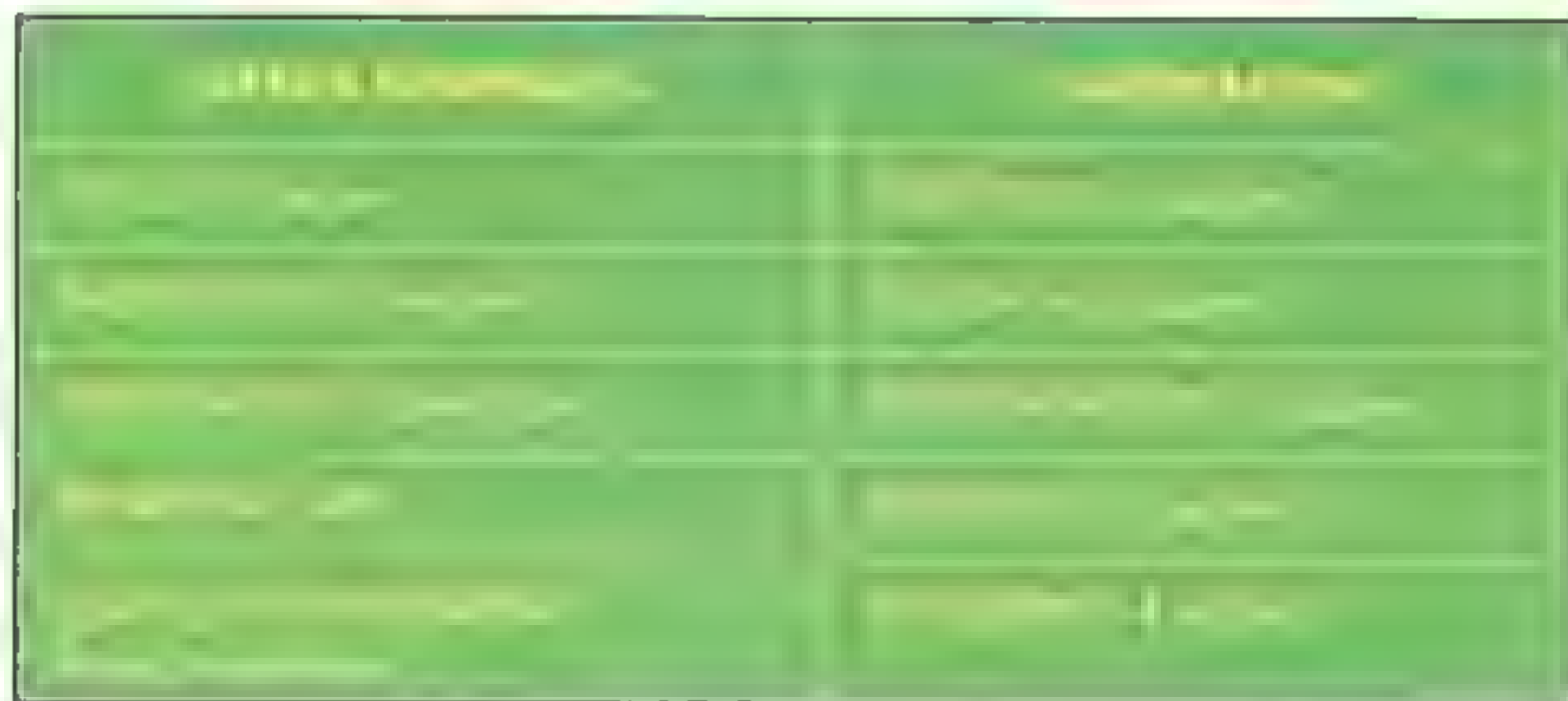
El protagonista de la aventura es **Isvar**, un intrépido príncipe que debe sortear los obstáculos que se le presenten, antes de que su energía se agote. Cuando ésta descienda peligrosamente es posible recuperarla

habitaciones un bruto que, con un palo en la mano, te perseguirá incansablemente hasta que lo destruyas o pases a otra estancia. Por último te recomendamos igualmente, tener mucho cuidado con un peligroso tornado y unas enormes esferas que pueden precipitar el final de la partida.

CINTURON NEGRO

Has abandonado la sala de meditación y te diriges al tatami, lugar donde se va a desarrollar el

combate. Un cinturón negro, décimo dan, está esperando. A la derecha un juez imparcial otorgará los puntos



que correspondan. El escenario del primer combate New York. Conviértete en un hábil karateca y podrás recorrer todo el mundo; cada vez que derrotes a un rival te trasladarás a otro país para realizar el siguiente combate.

Tras el ceremonioso saludo inicial comienza la pelea; vigila atentamente los movimientos de tu rival y al menor descuido, cuando baje la guardia, intenta derribarle.

Cada asalto tiene una duración máxima de treinta segundos, en los que debes alcanzar dos puntos para ganarlo. Puedes conseguir un *ippon* que significa un punto o un *waza-ari*

que supone medio punto, el juez decidirá si el golpe es merecedor de uno o medio punto. El combate se disputa al mejor de tres asaltos y si vences pasarás a enfrentarte con el rival sudamericano que tiene a Río de Janeiro como escenario.

El programa está realizado de tal forma que permite una gran variedad de movimientos, pudiendo saltar, avanzar, retroceder y girar; los golpes pueden darse con las manos o los pies, con patadas de pecho, de frente, laterales y todas las técnicas del karate.

En la pantalla se refleja en todo momento, por medio de círculos que

se encuentran en la parte superior izquierda y derecha, la puntuación de cada uno de los contendientes. En el medio figura el tiempo de asalto que resta y debajo del karateca que manejas aparece un flecha roja indicativa de la posición.

El programa admite dos jugadores y se puede conectar distintos *joysticks*. Pero su principal característica son los comentarios que realiza durante el combate. Efectivamente, cuando un karateca cae derribado el juez anuncia la puntuación y el color de quien se la adjudica. Aunque en un inglés algo tosco, es posible escuchar al juez a través del **Spectrum** dictaminando su veredicto. Esto constituye una auténtica novedad y lo convierten en uno de los pocos juegos del mercado que incorpora la voz. Además de ello, también incluye una pegadiza melodía con aire oriental que resulta muy agradable de escuchar.

Los aficionados a este bello deporte tienen ahora la ocasión de disfrutar del mismo sin correr ningún riesgo, y para aquellos que no lo son, este programa puede servir como una agradable introducción.

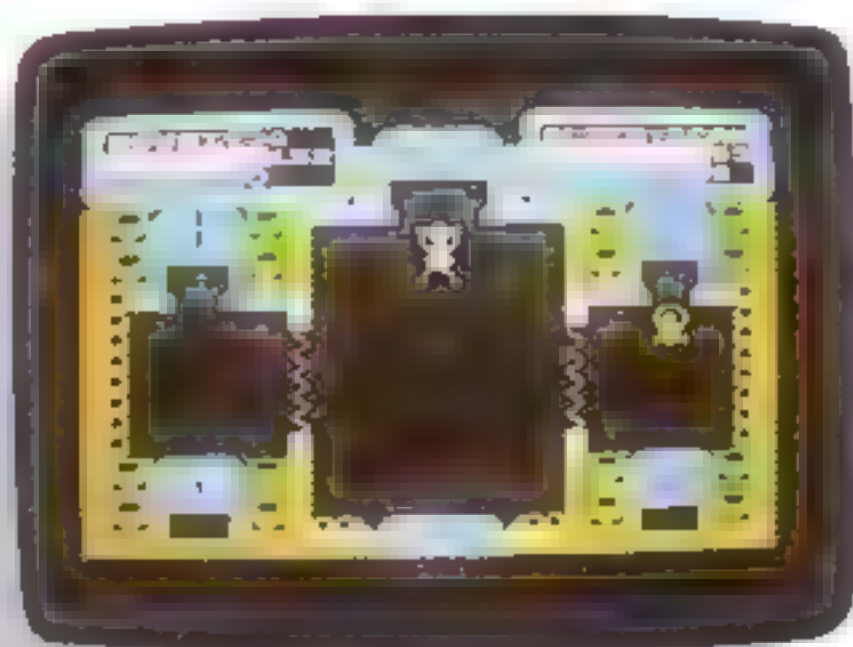
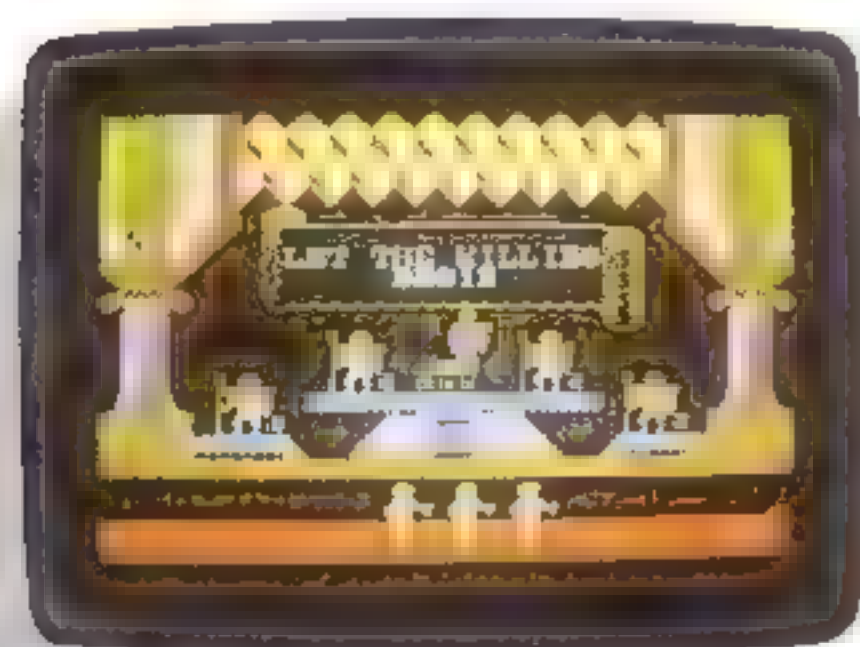
★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★

EL PERRO DE ESTRONCIO

Un grupo de malvados dictadores gobiernan la galaxia. Para imponer su ley han contrarado a unos cuantos asesinos a sueldo. El temor y el miedo reinan en todos los planetas. Los ciudadanos no pueden resistir más esta situación y deciden construir un mutante que limpie la galaxia.

¿Te gustaría participar en la aventura? Ahora puedes dirigir desde tu **Spectrum** los controles del mutante y encaminarlo por un complicado laberinto hacia el objetivo final: la destrucción de los peligrosos alienígenas. Los asesinos sin escrúpulos se han adueñado de la vida en la galaxia.

Cada uno de ellos se encuentra ubicado en un sector distinto y algunos tienen un comportamiento totalmente imprevisible. Tendrás que poner a prueba tus reflejos y moverte rápidamente para no ser alcanzado por sus mortales bombas. Pero no sólo debes vigilar a los alienígenas, sino que has de procurar



no rozar las zonas intermitentes que te causarían la muerte instantánea. Varios de los asesinos se han instalado detrás de una barrera electrificada, que si llegas a tocar será mortal y ándate con cuidado porque sus bombas rebotan en las paredes, multiplicando peligrosamente sus efectos. Otros se esconden tras unos soportales, apareciendo y desapareciendo muy rápidamente, y por supuesto lanzando sus mortales bombas.

Sin embargo no todo van a ser dificultades. Tu mutante ha sido construido con la más moderna tecnología y cuenta con un efectivo laser, que permite hacer frente a cuantos asesinos salgan a tu encuentro. También dispones de tres electro-bengalas que debes utilizar en cada uno de los **Medi-centrer** que alcances.

Inicialmente existen 93 asesinos, que deberás ir aniquilando con tu hostilidad e inteligencia para convertirte en un cazarecompensas al



que todas las galaxias agradecieran sus desvelos. Como pequeño consejo te podemos decir que cuando llegues al campanario aparecerá el horrible **Guerrero de Acero** para cuya destrucción deberás pulsar el botón amarillo, con lo que evitarás un gran peligro.

Este programa creado por **Premium Software** y **Quicksilva** presenta buena calidad gráfica, manejo sencillo y una distribución de teclas, además de la posibilidad de utilizar *joysticks*

para su control.

Las dificultades para conseguir realizar la misión son enormes y las posibilidades de éxito casi nulas, pero alguien debe poner fin a esta situación insostenible y salvar a la galaxia de la tiranía de los dictadores.

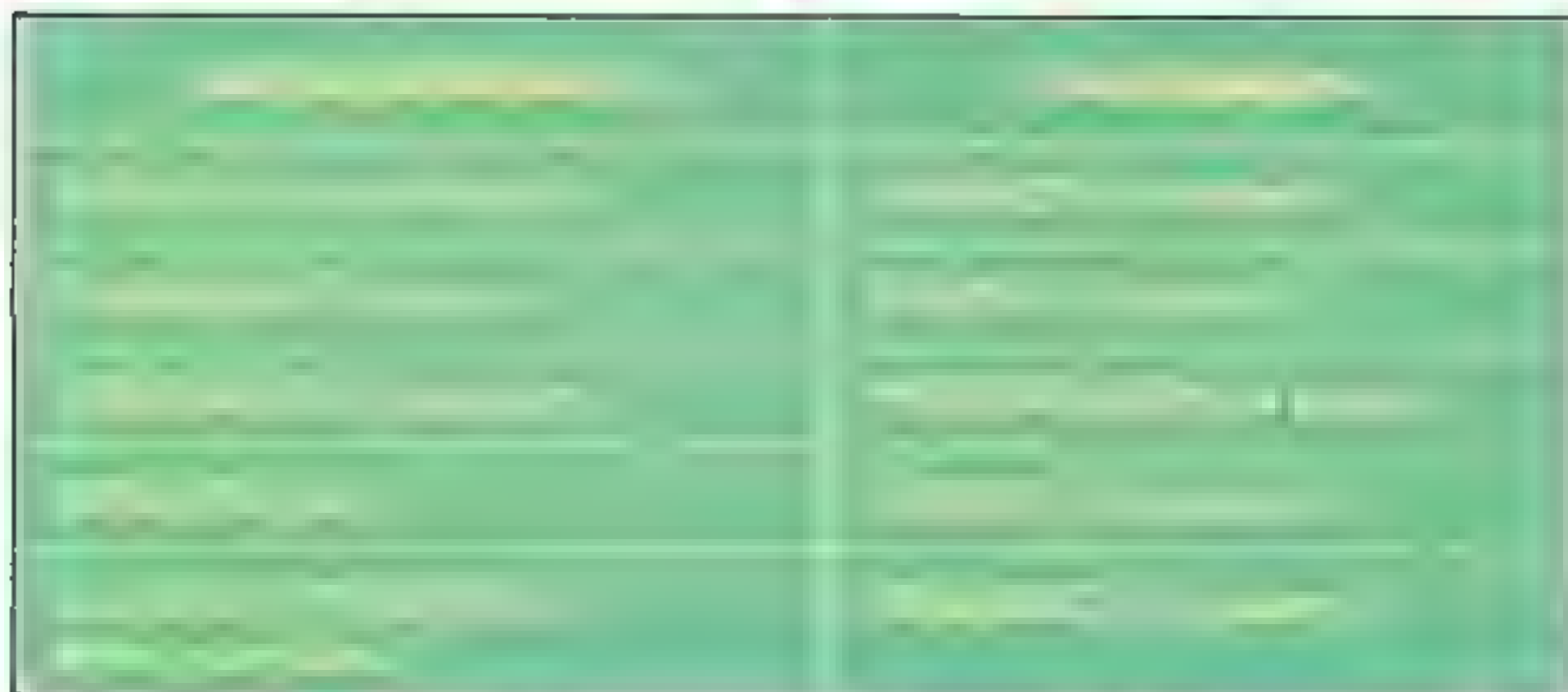
Debes poner a prueba tu capacidad y empeñar todos los esfuerzos en destruir las fuerzas del mal que se han adueñado de los planetas; el futuro de la galaxia está en tus manos, no la defraudes.

OOLONG, NINJAFAN FAN, TONFUN Y SUS AMIGOS

¿Te atreverías a enfrentarte tu solo a los nueve especialistas de **Kung-Fu** más peligrosos del mundo? Pues esa es la tarea que debe afrontar el joven **Oolong** para, emulando a su padre, llegar a ser nada menos que ¡Gran Masestro de **Kung-Fu**!, título que muy pocos pueden enorgullecerse de poseer. Para ello

deberá vencer en peligrosísimos combates al enorme **Buchu**, la bella **Star**, el maestro **Nuncha**, a **Pole** que le atacará con una vara, **Club** y su terrible escudo, a **Sword** especialista en el manejo de la espada, a la experta en **Ninjafan 'Fan**, a **Tonfun** y por último a **Blues** el Maestro de **Kung-Fu**; todos ellos fabulosos

especialistas en artes marciales que pondrán a prueba su habilidad, valor y rapidez de movimientos. Si te atreves a asumir el papel de **Oolong**, tienes oportunidad de hacerlo con «**Yie ar Kung-Fu**» juego de artes marciales, cuyo final objetivo es vencer al gran «**Blues**» Maestro de **Kung-Fu**, al que tendrás



de la pantalla, si esto sucede, salta por encima de tus enemigos y aléjate para preparar una estrategia de ataque, sabiendo que puede utilizar 16 distintos tipos de golpes, según lo aconsejan las circunstancias.

Si a pesar de hacer todo esto, tus enemigos son más hábiles y vencen, no te desanimes porque te quedan otras cuatro vidas para intentarlo. Además, si aplicas tus golpes certeramente, irás sumando puntos y al llegar a 20.000 tendrás una vida extra.

Este juego, que por lo demás es muy similar a los de artes marciales que ya existen, es entretenido sin más ambiciones aunque quizá un poco complejo de manejar si no usas *joystick* por la gran cantidad de teclas que has de utilizar.

en la pantalla solamente si antes has derrotado a otros ocho enemigos, cada uno con formas de luchar distintas. Para conseguir esto habrás de tener presente tres cosas: 1.º Has de encontrar el punto débil de cada

uno y golpearle en él. 2.º Algunos contrincantes te atacarán con armas y a éstos debes golpearles y retirarte rápidamente, para que no te alcancen con ellas. 3.º No te dejes acorralar nunca en uno de los lados



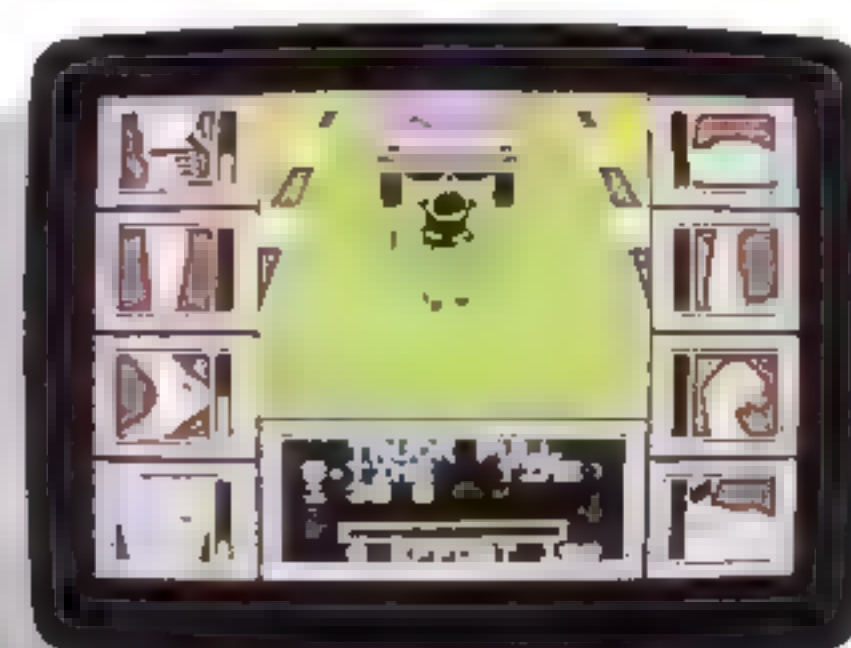
ENTRENA TUS MUSCULOS

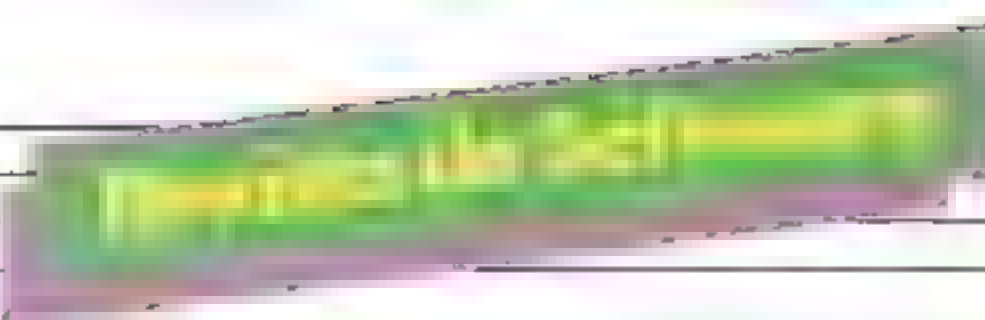
Poseer unos músculos muy desarrollados no es fácil. Son muchas las horas necesarias de dedicación intensiva y pocos los excesos alimenticios que se pueden cometer. Pero tan importante como

conseguirlos es saberlos utilizar correctamente.

En este programa, además de las sesiones de entrenamiento en las que debes poner a punto ocho partes del cuerpo, también tienes que utilizar

adecuadamente los músculos correspondientes: bíceps, flexores, aductores, gemelos, etc. El forzado Geoff Capes tiene las condiciones necesarias para convertirse en el hombre más fuerte del mundo, pero

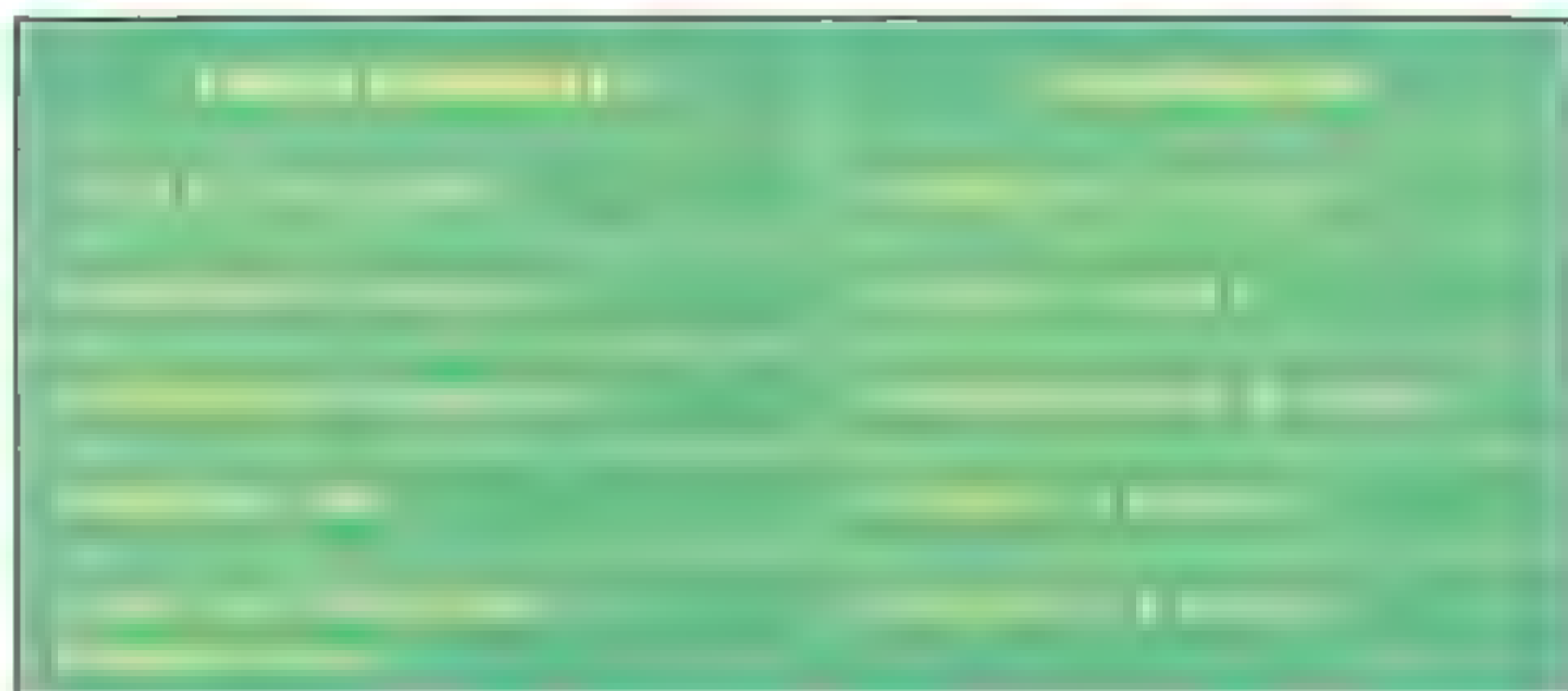




necesita alguien que dirija sus entrenamientos y le proporcione la habilidad precisa para llevar a cabo las diferentes pruebas por la que debe atravesar.

En total son seis las pruebas que ha de afrontar y ante cada una de ellas hay que realizar una sesión de entrenamiento en la que debe ejercitar los músculos para alcanzar la forma necesaria de cada uno de ellos y, prioritariamente, de aquellos que más se van a utilizar en la siguiente prueba.

El programa se inicia con la sesión de entrenamiento en la que repartirás la fuerza obtenida, entre las ocho partes. Tras ello la primera prueba: arrastrar un automóvil. En pantalla aparece nuestro protagonista tirando del automóvil, mientras en los laterales figuran los ocho músculos y mediante un recuadro amarillo se indica qué músculo es necesario utilizar para poder mover el coche. Con la mayor rapidez posible debes dirigirte hacia el músculo indicado y activarlo, de este modo **Geoff** podrá tirar del automóvil hasta llegar a la meta. Cuanto antes realices la prueba más



puntos obtendrás, pero recuerda que el tiempo máximo que dispones es de un minuto y medio.

La segunda prueba consiste en cortar un voluminoso tronco en un tiempo máximo de minuto y medio.

A continuación aparecerá un automóvil que debes levantar hasta conseguir ponerlo lateralmente.

La cuarta prueba combina la fuerza con la habilidad y rapidez. Nuestro amigo **Geoff** debe transportar cinco pesados barriles hasta un camión próximo en tan sólo dos minutos.

La siguiente prueba es muy típica de

las ferias y parques de atracciones; consiste en golpear fuertemente una plataforma y conseguir que suene la campana instalada en el extremo final.

En la prueba final tendrás que derrotar a un peligroso contrincante frente al que deberás desplegar toda tu fuerza e ingenio para conseguir proclamarte el hombre más fuerte del mundo. El programa ha sido desarrollado por **Martech** y, además de su originalidad, también hay que destacar que ha sido supervisado por el propio **Geoff Capes**.

LA MONTAÑA RUSA

¿Te apetecería pasar un divertido día en el parque de atracciones?. No tienes que pagar entrada y además puedes montar en todas las atracciones disponibles sin tener que pedir permiso a nadie. Pues si te animas, adelante, disponte a disfrutar de un inolvidable paseo por la feria, con el **Roller Coaster de Elite**.

Nuestro protagonista es un sujeto temerario que no se asusta ante ninguna dificultad por peligrosa que ésta sea; es capaz de montarse en cualquier atracción a pesar del riesgo que muchas de ellas entrañan.

El objetivo del juego es recolectar unos objetos, todos ellos iguales, que se encuentran ubicados en cualquier lugar del recorrido. Cada uno que

consigas significará cien puntos y hay que intentar obtenerlos todos.

El personaje puede desplazarse a izquierda y derecha, así como saltar y acelerar el paso por una feria que está repleta de atracciones: casetas de golosinas, cafeterías, bares, tivovivos, norias, montañas rusas, etc.





En su recorrido por el parque puede encontrar una gran montaña rusa en la que debe intentar atrapar todos

los objetos que surjan en su camino, eso sí, sin bajarse del coche en el que va, por lo que es necesario saltar

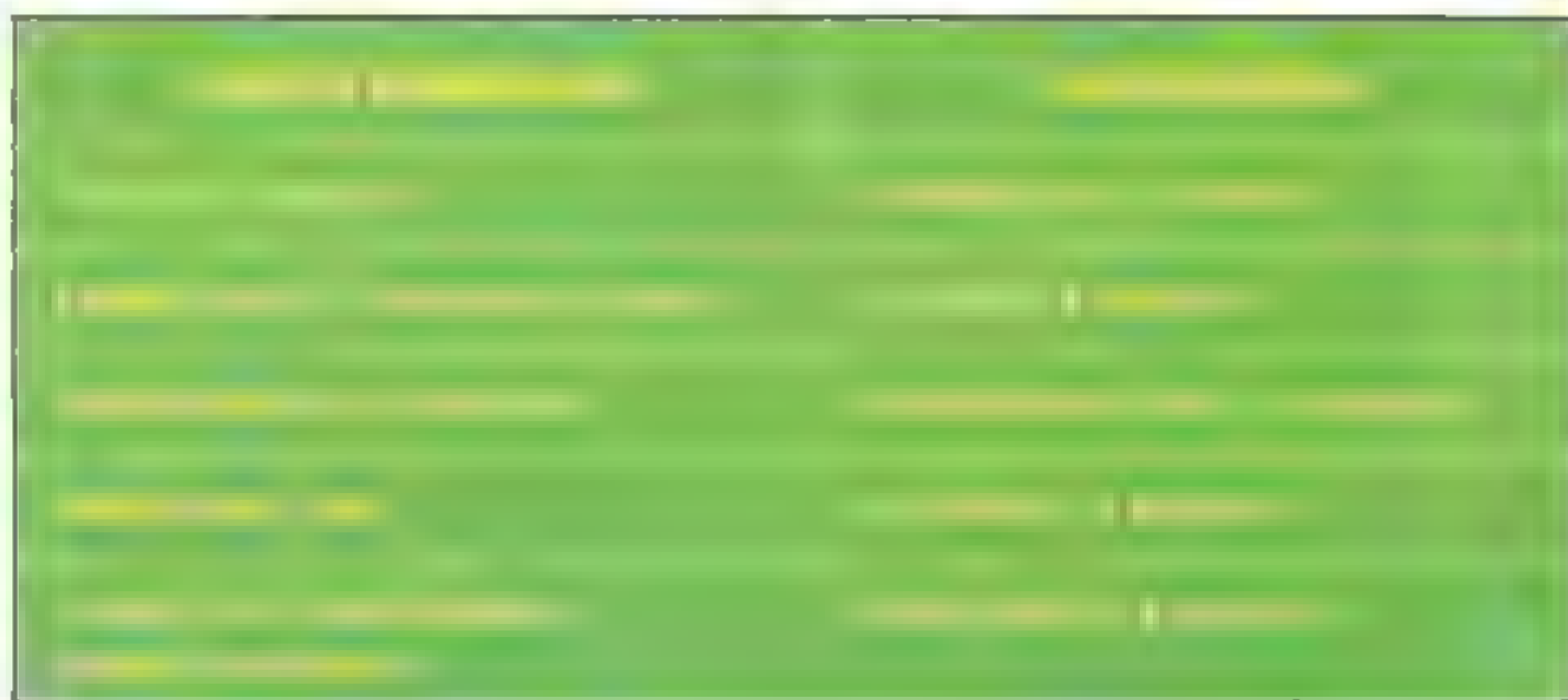
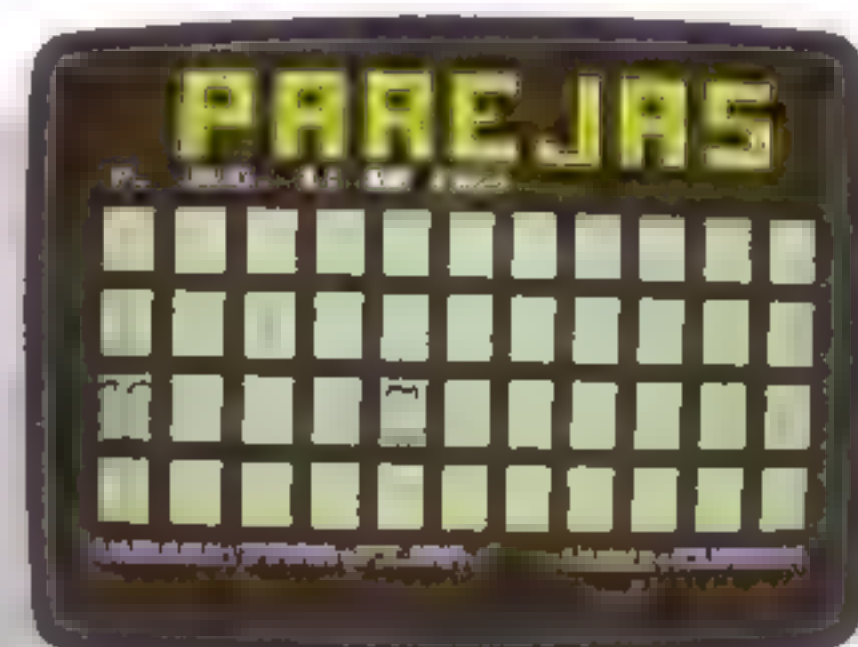
y volver a caer en el vehículo en marcha. Algo similar ocurre si consigue llegar a las minas de oro, donde montado en un vagón, debe alcanzar el extremo opuesto consiguiendo el mayor número de objetos posibles

También hay unos vertiginosos tiovivos cuya velocidad dificulta mucho el poder subirse en ellos, y una vez instalado, el mantenerse sin salir despedido.

Por si fuera poco también puede viajar en globo, ya que es otra de las atracciones que incluye esta feria, que tiene hasta un tren fantasma.

PAREJAS

Parejas es una producción española de **Software Center**, especialmente pensada para los aficionados a los «solitarios» y los juegos de baraja. El programa nos invita a elegir entre tres formatos distintos de cartas y a seleccionar el nivel de dificultad y el número de jugadores (uno o dos). A partir de este momento comienza el juego, poniendo a prueba nuestra memoria y capacidad de atención en una carrera por lograr agrupar por parejas todas las cartas, colocadas boca abajo, en el menor tiempo posible. La dificultad estriba en que, después de levantar dos cartas, ambas vuelven a ocultarse, de forma que debemos recordar su posición para no levantarlas de nuevo en tanto no las necesitamos para completar una pareja.



NOTA PARA LOS LECTORES DE MURCIA

El número del pasado mes de diciembre no fue distribuido Murcia y su zona de influencia, debido a un lamentable accidente que sufrió el camión que transportaba los ejemplares.

Se ha realizado un nuevo envío y durante los próximos días podréis localizarlos en el k.osco. De todas formas, también podéis pedirlos por correo a nuestro servicio de ejemplares atrasados.

Así pues, **Parejas** no es sólo un juego de cartas como otros existentes en el mercado, sino también un excelente ejercicio de memoria e inteligencia apto «para todos los públicos», y especialmente indicado para desarrollar la capacidad de atención y retentiva de los más pequeños. No obstante, el grado de interés que ofrece es bajo y acusa cierta falta de elaboración.



¡NO TE PIERDAS NI UN SOLO EJEMPLAR!

INPUT SINCLAIR quiere proporcionar a sus lectores este nuevo servicio de ejemplares atrasados para que no pierdan la oportunidad de tener en sus hogares todos los ejemplares de esta revista, líder en el mercado español.

A partir de este mes podréis solicitar

cualquier número de INPUT SINCLAIR que queráis, siempre al precio de cubierta (sin más gastos).

Utiliza el cupón adjunto, enviándolo a **EDISA** (Dpto. de Suscripciones), López de Hoyos, 141 - 28002 Madrid, o bien llámanos por teléfono al (91) 415 97 12.



**siempre a
tu servicio**

CUPON DE PEDIDO

SI, envíenme contrarreembolso ejemplares de INPUT
SINCLAIR de los números:

(marca con una (X) tu elección)

1

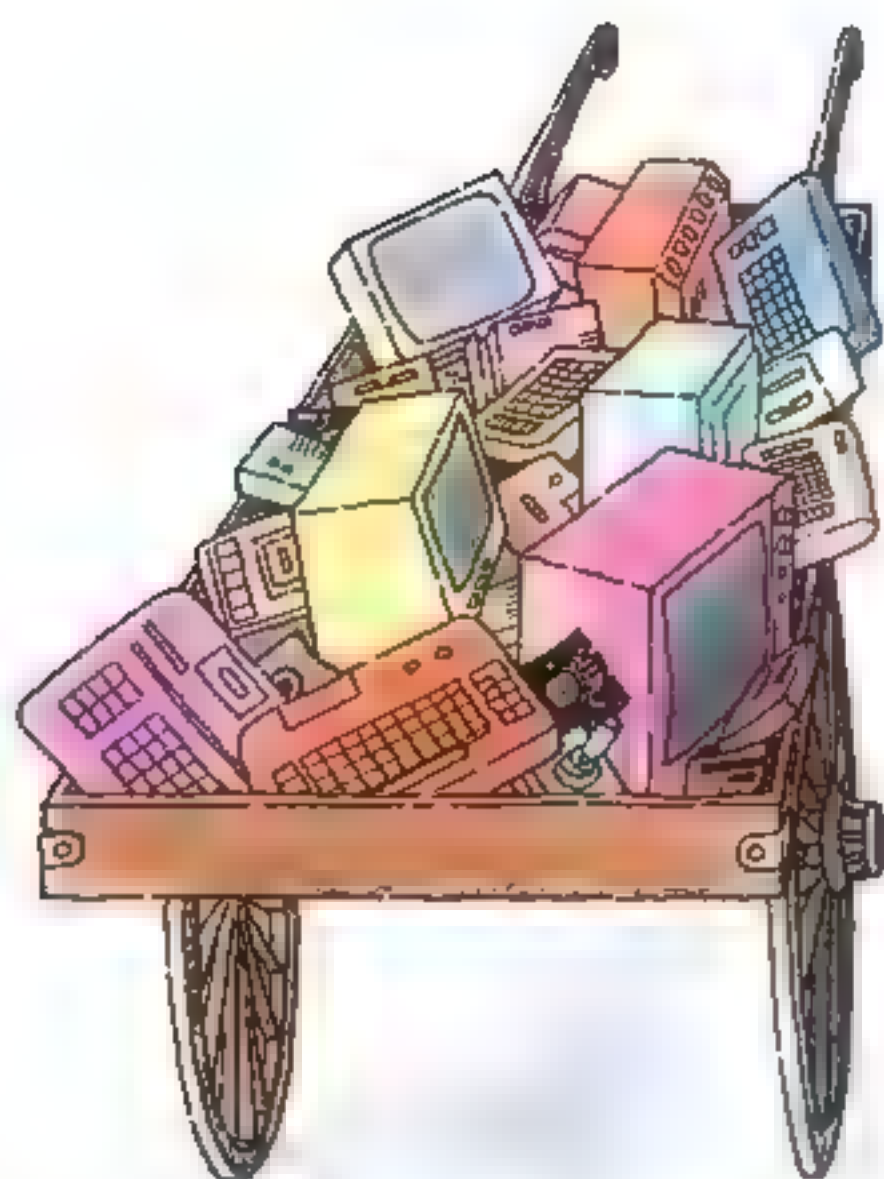
2

3

4

5

NOMBRE _____
APELLIDOS _____
DOMICILIO _____
NUM. _____ PISO _____ ESCALERA _____ COD. POSTAL _____
POBLACION _____ PROV _____
TELEFONO _____ FIRMA _____



EL ZOCO

Compro ordenador cualificado de precio asequible (Sinclair).

Jesús Angel García Márquez.
Juan de Garay, Bl. A, 2.
Almería.

• • • • •

Interesados en cambiar ideas, programas o trucos para el Spectrum 48 K.

Luis Manuel Querol García.
Escultor Querol, 2.
Uldecona (Tarragona).

• • • • •

Intercambio trucos, ideas, etc. sobre el ZX Spectrum.

Manel.
Menéndez Pelayo, 18 - 4.º, 2.º.
Teléf. 815 53 26.
Vilanova i la Geltrú (Barcelona).

• • • • •

Cambio/vendo programas Spectrum 16 K y 48 K. Sólo tardes. Preferentemente Madrid capital.

Joaquín.
Teléf. 416 59 80.

• • • • •

Se cambia videojuego para televisión con 4 juegos y los mandos, por joystick más interface Kempston o cualquier otro periférico. También se vende por 4.000 pts.

Manuel Antonio Conesa Costa
Alfonso X el Sabio, 47 - 8.º C.
Teléf. (928) 53 34 32.
Cartagena (Murcia).

• • • • •

ZX-Spectrum Plus, nuevo (procede de sorteo y yo ya tenía uno) y completo con cables, etc. + 2 cintas de regalo. Mínimo 30.000 pts. (Tiene garantía vigente hasta enero-86.)

Alfonso Casas Aós.
Avda. Arias Maldonado, 11 - 4.º F.
Teléf. (952) 82 05 41 - 82 58 55.

• • • • •

Vendo Spectrum 48 K con todas las conexiones. Manual en inglés y castellano y fuente de alimentación. Todo por 25.000 pts.

Josep M.ª Caixas Fernández.
Albert Coto, 2.
Teléf. (972) 50 50 48.
Figueras (Girona).

Me gustaria ponerme en contacto con usuarios de Spectrum, para formar un club en León.

Oscar.
Teléf. 23 14 14.
León.

• • • • •

Cambio por software, o compro, instrucciones en castellano de los programas para Spectrum: Cosmic Cruiser, Atic Atac, Pedro, Psytron, Fantastic Voyage, Skool Daze, Monty is Innocent, Strange Loop y Shadowfire. (Sólo de Madrid.)

Miguel.
Teléf. 415 93 79.
(De 15.30 a 18 h.).
Madrid.

• • • • •

Compro joystick con interface para Spectrum 48 K. Precio a convenir. También me interesa hacer un club de usuarios del Spectrum.

Diego Enrique Alarcón Aguirre.
Teléf. (951) 43 10 02.
Almería.

• • • • •

Vendo videojuego Philips (Videopac+ G7400) poco utilizado, casi nuevo. Precio 15.000 pts. (El precio incluye 4 cintas de juegos valoradas todas juntas en más de 10.000 pts.).

Guillermo Coll Florit.
Arqueros, 11 - 4.º B.
Teléf. 218 70 67.
28024 Madrid.

• • • • •

Vendo ZX Spectrum 48 K, cables, adaptador 9 V y manuales en castellano por 22.000 pts. Contrarreembolso.

Juanjo González.
Pdo. 6.
Teléf. (922) 33 39 65.
La Orotava (Tenerife).

• • • • •

Desearia intercambiar programas para el Spectrum 48 K, con lectores de INPUT.

Oriol Domingo.
Vila Cinca, 166 - 2.º.
Teléf. (93) 716 81 67.
Sabadell (Barcelona).

• • • • •

Vendo barata impresora térmica «Alphacon» timex en perfecto estado.

Josepxo.
Teléf. (943) 21 37 96.
San Sebastián.

• • • • •

Vendo ZX-81 con 16 K, todos los cables y manual, casi sin usar, regalo libro de juegos y cassette de juegos. Precio: 8.000 pts.

Carlos.
Teléf. 12 46 47.
Granada.

• • • • •

Vendo ordenador personal de bolsillo Casio PB-100. Recién comprado. Regalo libros de instrucciones en castellano. 10.000 pts.

José M.ª
Teléf. (972) 50 50 48.
Figueras (Gerona).

• • • • •

Vendo ZX Spectrum 48 K RAM, con salida para monitor, incluyendo cables, manuales en castellano, 2 libros de programación en Basic y 10 cintas de juegos de regalo por la cantidad de 27.000 pts.

Juan Carlos González.
Hortensia, 7 - Bajo D.
Teléf. (91) 675 62 43.
Torrejón de Ardoz (Madrid).

• • • • •

Compro libros Spectrum. Mandar lista. Mandar precios.

Francisco González Batlle.
Muntaner, 597 - 6.º, 1-A.
08022 Barcelona.

EL ZOCO

Se intercambian programas para ZX Spectrum. Preferentemente de 48 K. Poseo últimas novedades inglesas.

Francisco Javier Olmos.
Alfonso VI, 64 - 2.º izqda.
Teléf. 947) 31 14 52 (de 7.30 a 11.30).
Miranda de Ebro (Burgos).

• • • • •

Vendo Spectrum de 48 K en perfecto estado con todos sus cables y manuales de iniciación. Con interface joystick RAM TURBO válido para todos los sistemas y entrada para cartuchos y joystick quick shot y 100 programas de los mejores como Dam Buster, Alien 8, El Grand National, etc. Por 45.000 ptas. negociable. Vendo el mejor programa de Karate para el Spectrum el Exploding Fist por 1.000 ptas.

Jorge Morales Gutiérrez.
Cosmógrafo Ramírez, P/15.
Teléf. (96) 288 36 08.
Xátiva (Valencia).

• • • • •

Vendo los programas originales: Chess, Horace goes skiing, Hormigas, Pánico en el express, Chequered Flag, Survival, Scrabble, Make-A-Chip, Trans América. Todo ello por 4.500 ptas.

Francisco Javier.
Teléf. (968) 84 23 36.

• • • • •

Vendo Spectrum 48 K en perfecto estado. Incluye cables, cinta horizontes y manuales en inglés y castellano. Regalo cinta con diez juegos. Urge. Todo por 25.000 ptas.

Rafa.
Teléf. 465 03 86.
Madrid.

• • • • •

Desearía que algún amable usuario de ordenador ZX Spectrum me proporcionara la enseñanza de grabación en cinta, por teléfono o carta. Pagaría hasta 6.000 ptas.

Iñigo.
Mendietas Zeharbide, 6 - 2.º A.
Teléf. (94) 676 02 55.
Sopelana (Vizcaya).

• • • • •

Cambio por impresora Seikosha gp-50 S o Interface 1+Microdrive, un amplificador Sony de 185 W stéreo y un

Psicodélico de 3 canales (ambos con poco uso), o bien los vendo por 23.000 y 7.000 ptas respectivamente.

Paco.
Teléf. (922) 28 44 03.

• • • • •

Compro cinta de Horizontes por 3.000 ptas. ya que con mi ordenador no venía.

Iñigo.
Mendietas Zeharbide, 6 - 2.º A.
Teléf. (94) 676 02 55.
Sopelana (Vizcaya).

• • • • •

Vendo ordenador ZX Spectrum 16 K, comprado en agosto del 84, con manuales, cables y una cinta de 60 minutos con juegos; todo esto por 20.000 ptas. tratables.

Alberto.
Teléf. 478 15 62
(a partir de las 19 h.).
Madrid.

• • • • •

Vendo impresora ZX Printer muy barata, perfecto funcionamiento, interesados llamar a:

José Antonio Suárez Márquez.
Vázquez Varela, 19 - 3.º B.
Teléf. (986) 41 18 95.
Vigo-4 (Pontevedra).

• • • • •

Vendo e intercambio cintas de juegos de ZX Spectrum 48 K.

Manuel Magalotti García.
Teléf. 733 00 58.
Madrid.

• • • • •

Vendo ZX 81 ampliado a 64 K con manual en castellano y demás accesorios por 15.000 ptas.

Alfonso Abellán Castillo.
Rosellón, 51 - 5.º, 3.º.
Sant Boi de Llobregat (Barcelona).

• • • • •

Vendo ordenador ZX Spectrum (buen funcionamiento), con cables, manual, etc. + grabadora + joystick con Interface Program + Numerosos juegos y programas + Revistas. 45.000 rebajables.

Alberto Peón Souto.
Jerónimo Ibrán, 4 - 3.º A.
Teléf. (985) 29 19 15.
33001 Oviedo.

• • • • •

Necesito correspondencia con usuarios del ZX Spectrum para que me resolvieran muchas dudas e intercambio de ideas.

José Miguel Llop Juan.
Primavera, 8 - 3.º.
Hospitalet de Llobregat (Barcelona).

• • • • •

Intercambio y compro programas de juegos, utilidades, educativos, etc., para el ordenador ZX Spectrum de 16 K y 48 K. Interesados enviar lista de programas a:

José Ramón Román Collado.
Avda. de la Raza, 1 - D-7.º.
41012 Sevilla.
(Prometo contestación a quien escriba.)

• • • • •

Desearía intercambiar juegos actuales con poseedor de Dam Busters y otros de guerra.

Francisco Javier Mora Sánchez.
Sierra Bermeja, 10 - 1.º A.
29003 Málaga.

• • • • •

Cambio juegos tiro «Arcade» y juegos de las principales marcas (Psion, Virgin, Load N'Run, Ultimate, Durrell, etc.) por programas de gestión y contabilidad. Preferiblemente sean usuarios que vivan en Barcelona ciudad.

Javier López Ortí.
Martí, 140, 3.º.
Teléf. 214 46 66.
(No llamar después de las 21.30).
08024 Barcelona.

• • • • •

Intercambio programas, ideas, para el Spectrum. Últimas novedades. Enviar lista.

Rafael Antón Sánchez.
Aptdo. de Correos 451.
Elche (Alicante).

• • • • •

Quisiera contactar con chicos/as de Barcelona para intercambiar programas, ideas, trucos...

Agatha.
Teléf. 239 89 23.

Ayúdanos a ayudarte

Puedes ganar un SPECTRUM PLUS



Encuesta para conocer tu perfil y preferencias

Es muy difícil que podamos hablar con todos vosotros para conoceros mejor y saber cuáles son vuestras preferencias. Para tener constancia de opiniones y necesidades hemos de recurrir a las frías encuestas y dejar que se pronuncie la estadística. Los beneficiados seremos todos: vosotros porque nos decís qué queréis y nosotros porque al haceros caso continuaremos haciendo vuestra revista favorita.

1. ¿Tienes ordenador? Sí ☐ No ☐
2. En caso afirmativo, ¿Cuál?
3. ¿Cuánto tiempo hace que lo posees?
4. ¿Qué periféricos y accesorios tienes?
5. ¿Desde qué número lees INPUT? N.º:
6. ¿Cómo conociste INPUT?
7. ¿La compras en kiosk? ☐ ¿Eres suscriptor? ☐
8. ¿Con qué frecuencia la adquieres?
Todas ☐
De vez en cuando ☐
Otros ☐
9. ¿Cuántas personas leen tu ejemplar de INPUT?
10. ¿Lees toda la revista? Sí ☐ No ☐
11. ¿Tecleas?:
Todo ☐
Casi todo ☐
Alguna cosa ☐
Nada ☐
12. ¿Califica qué secciones te gustan más con mayor puntuación (10 a 0):
Programación
Código máquina
Robótica
Coleccionable
Revista de *hardware*
Revista de *software*
Aplicaciones
Actualidad
Libros
Buzón
Zoco
Educación
Los mejores de INPUT
13. Califica las portadas entre 10 y 0, según lo que te hayan gustado:
- N.º 1
- N.º 2
- N.º 3
- N.º 4
- N.º 5
- N.º 6
14. ¿Qué artículo te gustó más del ejemplar?:
N.º 1
- N.º 2
- N.º 3
- N.º 4
- N.º 5
- N.º 6
15. ¿A qué te dedicas cuando no estás con el ordenador?, ¿estudias o trabajas?, ¿en qué, si no es indiscreción?:
16. ¿Qué secciones o temas echas de menos en INPUT?:
17. ¿Estás satisfecho con tu ordenador? Sí ☐ No ☐
18. ¿Estás contento con los servicios técnico y postventa? Sí ☐ No ☐
19. ¿De qué marca y modelo será el próximo ordenador que compres?
20. ¿Piensas cambiarlo pronto? Sí ☐ No ☐
21. ¿Aproximadamente, cuántos programas comerciales forman tu biblioteca?
22. ¿Cada cuánto tiempo compras uno nuevo?
23. ¿Lees otras revistas de ordenadores? Sí ☐ No ☐
24. En caso afirmativo, ¿Cuáles?
25. ¿Por alguna razón en particular?
26. ¿Quieres añadir alguna cosa más?

Con solo contestar a esta encuesta puedes ganar un SPECTRUM PLUS o su valor equivalente en accesorios y/o software..

Así, que escribe tus datos para participar. La fecha límite es el próximo 5 de abril. No es preciso que contestes las preguntas para las que no tengas respuesta. Tampoco es necesario que las respuestas sean favorables a nuestra labor para que puedas participar en el sorteo. Critícanos si honestamente crees que debes hacerlo. Envíanos este cuestionario o su fotocopia a: ENCUESTA INPUT SINCLAIR, c/ Alberto Alcócer, 46-4.º 28016 MADRID.

Nombre Apellidos
Edad Dirección Dto. Postal
Teléfono de contacto Gracias y suerte con el premio.

Almacenes de Componentes, Su problema de libros de Electrónica...



RESUELTO

ELECTRO-LIBRO

Es una sección de **HIPERMUSIC**
Barquillo, 21. Telfs. 221 72 01 - 232 59 16
28004 MADRID

Contacte con nosotros ya.

Deseo recibir el catálogo de títulos y precios de los libros distribuidos por ELECTRO-LIBRO.

NOMBRE APELLIDOS

CALLE N.º D.P.

CIUDAD PROVINCIA

Descuentos especiales para mayoristas y comerciantes
servicio de novedades para todos nuestros clientes.

Primero fué **SUMMER GAMES**
despues **SUMMER GAMES II**

y ahora...

WINTER GAMES



Ha conseguido el oro en los Juegos de Verano y Juegos de Verano II. ¡Ahora estamos en los Juegos de Invierno! y qué increíble marco, un completo país de invierno realizando seis competiciones de acción. Puede competir contra sus amigos o el ordenador. Primero elija el país que quiera representar. Practíquelo, prepárese y aprenda una estrategia para ganar en cada competición. Ahora comience la ceremonia de apertura y la competición. ¿Será usted quien consiga el oro en la ceremonia de entrega de premios? La búsqueda del oro continúa... y está todo aquí: la estrategia, el reto, la competición, el arte y la pompa de los Juegos de Invierno.

- Seis competiciones de invierno: Bobsled, salto de ski, patinaje artístico, patinaje libre estilo, Hot Dog Aéreo y el ski de fondo.
- Ceremonias de apertura, cierre y entrega de premios con himnos nacionales.
- Compita contra el ordenador o contra sus amigos o familia.
- Control único por el joystick, necesita destreza y cronometraje.
- Uno a ocho jugadores.

EPYX
COMPUTER SOFTWARE

Fabricado y distribuido en exclusiva por

COMPULOGICAL S.A.
Calle de la Mariposa, 25 - 28015 Madrid - Tel. 241 1040

Distribuido en Cataluña y Baleares por: **YA ESTA DISPONIBLE PARA EL SPECTRUM**
DISCLU, S.A. - Balmes, 58 - BARCELONA - Tel. (93) 302 39 08 - P.V.P. 2.300 Ptas.